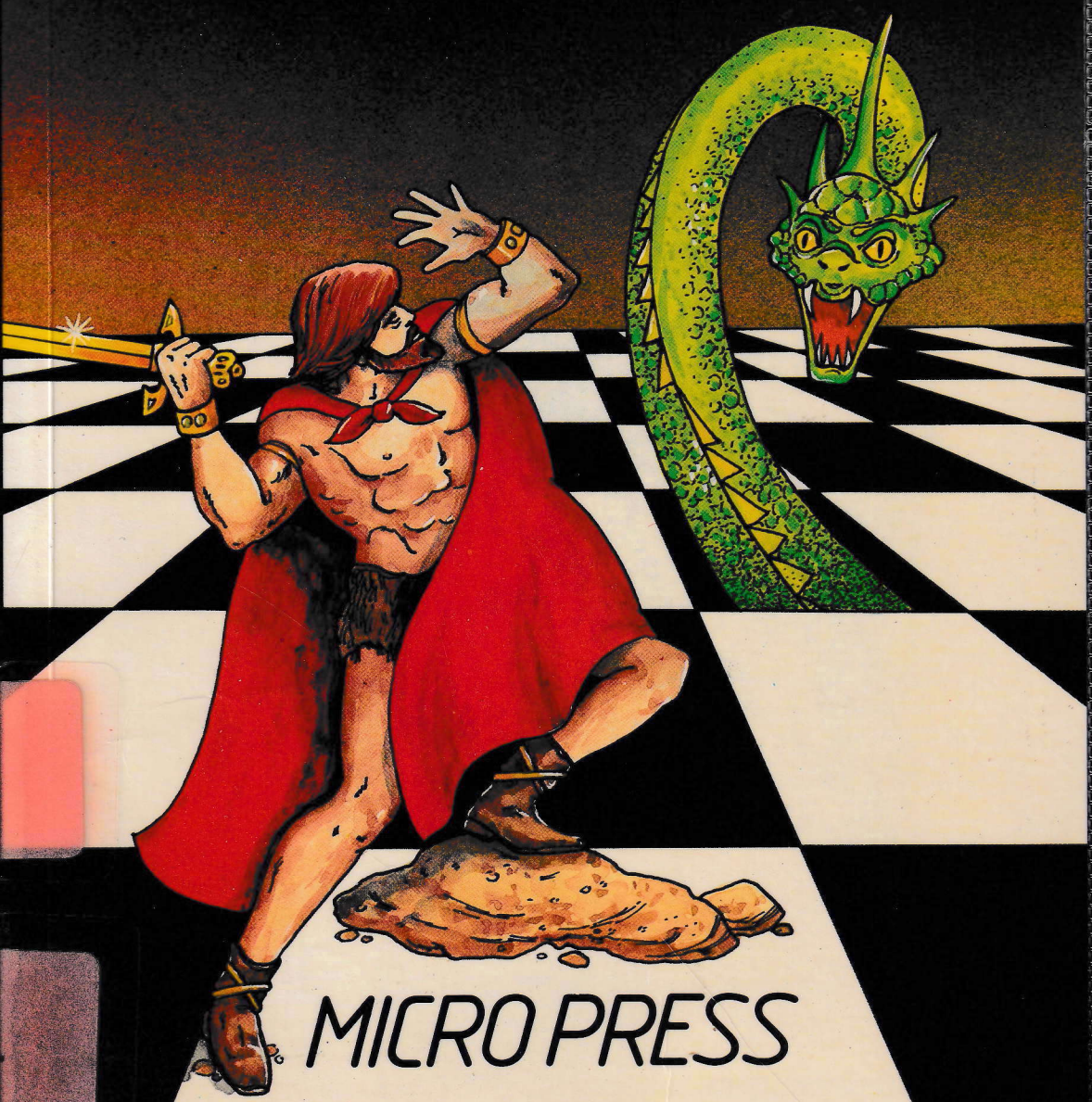
SPECTRUM SUPERGAMES

Richard G. Hurley



MICRO PRESS

Z 7447 0017 5

Spectrum Supergames

LONDON BOROUGH OF GREENWICH

Greenwich Libraries Book Sale

GREENWICH LIBRARIES

RESERVE STOCK (P)

LIB.116

GREENWICH LIBRARIES

3 8028 00852499 0

Spectrum Supergames

Richard G. Hurley

MICRO PRESS

SBN 1 7447 | 2 0017 | 3 5 | LOC. Z | COPY 1
EDITION DATE 1984.
CLASS NO. 794
INVOICE JM 13 NOV 1984 £5.95

First published 1984 by
Micro Press
Castle House, 27 London Road
Tunbridge Wells, Kent

© Richard G. Hurley 1984

All rights reserved. No part of
this publication may be reproduced,
stored in a retrieval system, or
transmitted, in any form, or by any
means, electronic, mechanical, recording
or otherwise, without the prior
permission of the publishers.

British Library Cataloguing in Publication Data

Hurley, Richard G.
Spectrum supergames.
1. Computer games 2. Sinclair ZX Spectrum (Computer)
I. Title
794.8'028'5404 GV1469.2

ISBN 0-7447-0017-5

Typeset by Keyset Composition, Colchester, Essex
Printed by Mackays of Chatham Ltd

To my mother
For all her care and attention over the years

Acknowledgements

I should like to thank James Mead, Graham Budd and Stephen Lacey for their valuable contributions towards completing this book. I am also particularly grateful to Christopher Turner for his excellent imagination and his skill in producing ideas for the illustrations, and to David Virgo for his programs and his help in correcting my English; without such help this book would never have been written. Finally I should like to thank my wife, Josie, whose encouragement and superb skill in the art of coffee-making have made all this possible.

Richard G. Hurley

Hassocks, 1984

Contents

Introduction

This book contains 13 graphic games, all of which are different. They include arcade games, games of strategy and games of skill.

Various techniques, including useful machine code routines, new character sets and methods of programming which will save both time and memory, are employed in the programs; wherever possible these methods are explained in a clear and concise manner. Consequently, this book provides instruction in the art of games writing, as well as endless hours of entertainment.

Each of the 13 games is divided into a number of sections as described below.

Description

A brief description of the game is given. The object of the game is outlined and the tasks that the player will be presented with are indicated.

Hints on Entry

As some of the programs are quite long and complex, it is highly likely that errors will be made during the typing stage. This section is intended to assist the user by pointing out the complex lines where errors are most likely to occur. It is therefore most important that constant reference is made to this section during the typing stage.

Techniques

Each program will be accompanied by a short section instructing the user about one of the more complex techniques employed in

the program. It is hoped that by reading these notes and referring to the listings valuable information about the methods employed will be gained. This knowledge could then be put to use in the design of more complex games.

Instructions

All the games in this book are designed to be playable without any written instructions. This section is included, however, and contains details about loading the games as well as a few outline instructions.

Listing

The last section of each chapter will contain the listing together with a notice similar to that shown below.

IMPORTANT
1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.
2) ALL GRAPHICS CHARACTERS ARE REPRESENTED BY LOWER CASE LETTERS.

As can be seen from the notice, all games are written in the CAPS LOCK mode, so that the graphics characters represented by lower case letters are easily recognisable. These graphics characters must be entered by using the graphics mode (see Spectrum manual).

Chapter 1

Fantastic Journey

Description

News has reached you that Dr Hans Zarkov, founder of the World Peace Organisation, has been rushed to hospital with a suspected brain tumour. A few hours later, this diagnosis is confirmed, and the doctor in charge of the case announces that, due to the position of the growth and the condition of the patient, major surgery is impossible. The only chance of survival for Dr Zarkov rests in the hands of the Ultra Subminiature Research Team.

As the captain of the vessel SS *Nautilus*, it is your responsibility to pilot your craft, from the patient's leg through his bloodstream to the brain so that your first officer can use the onboard laser to dissipate the cancerous growth.

At first, the job seems perfectly routine, the reducing process

works well, and you are injected into the leg without any problems. However, only minutes after your journey commences you realise that you are going to have oxygen and fuel problems, and that your only hope of survival and a successful mission is to collect supplies from the various vital organs encountered during your fantastic journey through the human body.

Hints on Entry

The first thing to note is that the program contains a large amount of data, which is required for the alternative character set used throughout the game. This data should be entered with great care as a missing or incorrect value is likely to result in garbage being printed on the screen.

As the program uses high-resolution perspective graphics, very little use is made of the user-defined graphics. Those that are used are located in line 1030 and are represented by small characters.

The other area where care is necessary is in the entering of the section relating to the map (lines 298–360). All of these lines should consist of 26 characters; the number of spaces should be carefully checked by comparison with the previous line.

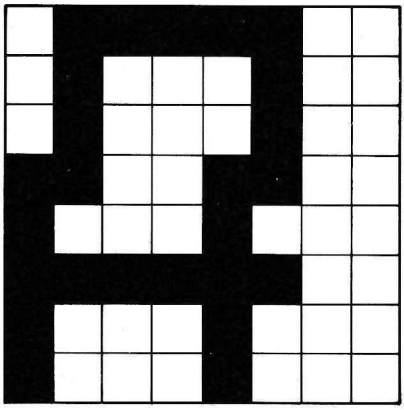
Techniques

When playing Fantastic Journey you will see that it is possible to create a special character set to be used to enhance the overall appearance of a game of this type.

If you are familiar with creating user-defined graphics then creating your own character set will not pose any problems. A character is constructed on an 8 × 8 grid (see Fig. 1) with each row being converted into binary and then into a decimal. The numbers contained in the right-hand column are then poked into consecutive memory locations where the character set is to be stored.

The original set used by the Spectrum is stored in ROM from location 15360 to location 16128, and because of the structure of ROM these values cannot be altered. The first step, therefore, in defining a new character set is to transfer the original from ROM

Letter A



01111100	124
01000100	68
01000100	68
11001100	204
10001000	136
11111100	252
10001000	136
10001000	136

Fig. 1

to some suitable location in RAM where it can be changed. (The top of memory from 63000 onwards is generally the best place.) The following program can be used to perform this transfer:

```
10 LET ST = 63000
20 FOR I = 0 TO 2040
30 POKE ST+I,PEEK(15360+I)
40 NEXT I
50 POKE 23606,24
60 POKE 23607,246
```

The last two lines are extremely important as they change the system variable CHARS, indicating to the computer the position of the new character set.

The character set has now been transferred from ROM into RAM, and we are now ready to start changing the characters by poking new values into the appropriate locations.

Example

Consider the problem of redefining the letter A to the design shown in Fig. 1. We must first decide where the representation of A starts within memory. This can be found using:

```
LET ST = (CODE(A) *8) + 63000
```

The new character can now be entered into memory by using the following short program:

```
 70  LET ST=(CODE(A)*8)+63000
 80  FOR I=0 TO 7
 90  READ N
100  POKE ST+I,N
110  NEXT I
120  STOP
130  DATA 124, 68, 68, 204, 136, 252, 136, 136
```

The letter A has now been redefined and will appear as such in all listings and printings. If it is required to define other letters (B, C, D, etc.), then we need only change the second value in the FOR loop and increase the amount of data accordingly.

Instructions

Load the game from tape to microdrive by typing:

```
LOAD""
```

or

```
LOAD *"M";1;"FANTASTIC"
```

When loaded the game will run automatically, although there will be a short delay while the new character set is defined. The game will then commence with your craft being injected into the patient's body. As the craft is computer controlled, most of the appropriate actions will be taken automatically, and it is your task to control the craft by using the cursor keys until your mission is successfully completed. Good luck and bon voyage.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED IN LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
  1 BORDER 0: PAPER 0: INK 7: BRIGHT 1:
 CLS : PRINT AT 10,4;"SETTING UP - PLEAS
E WAIT"
  2 PRINT AT 5,7;"FANTASTIC JOURNEY"
  3 POKE 23658,8
 10 FOR B=1 TO 13
 20 READ A$: FOR I=0 TO 7: READ A: POKE
 USR A$+I,A: NEXT I
 30 NEXT B
 100 DATA "A",3,12,49,66,144,160,146,137
,"B",223,56,128,0,48,35,7,7
 101 DATA "C",0,192,32,56,4,195,193,194,
"D",65,48,8,14,1,0,0,0
 102 DATA "E",135,19,56,197,0,0,0,0,"F",
194,4,24,224,0,0,0,0
 103 DATA "G",18,50,249,1,193,120,24,5,"
H",38,36,193,242,246,228,248,240
 104 DATA "I",0,127,255,127,31,3,0,0,"J"
,0,252,254,254,254,254,62,12
 105 DATA "K",56,68,162,170,138,146,132,
132,"L",4,26,18,43,44,16,17,14,"M",148,1
68,144,16,32,64,128,0
 110 FOR I=0 TO 768: POKE 64256+I,PEEK (
15616+I): NEXT I
 115 POKE 23607,250
 120 FOR I=0 TO 4: READ A: FOR C=0 TO 7:
 READ B: POKE A+C,B: NEXT C: NEXT I
 130 DATA 64504,126,70,6,30,24,0,24,0,64
736,64,96,48,24,12,6,2,0,64264,24,24,24,
24,24,0,24,0,64464,0,24,24,0,0,24,24,0,6
4488,0,62,62,0,62,62,0,0
 140 FOR I=64320 TO 64336: READ A: POKE
I,A: NEXT I
 150 DATA 0,12,24,24,24,24,12,0,0,48,24,
24,24,24,48,0
 160 FOR I=64352 TO 64374: READ A: POKE
I,A: NEXT I
 165 DATA 0,0,0,0,0,24,24,48,0,0,0,126,1
26,0,0,0,0,0,0,0,0,24,24,0
```

```
 170 FOR I=64521 TO 64729: READ A: POKE
I,A: NEXT I
 180 DATA 60,36,36,126,98,98,98,0,124,68
,68,126,98,98,126,0,126,66,66,96,96,98,1
26,0,126,66,66,98,98,98,126,0,126,64,64,
126,96,96,126,0,126,64,64,126,96,96,96,0
,126,66,64,102,98,98,126,0,66,66,66,126,
98,98,98,0,16,16,16,24,24,24,24,0
 184 DATA 4,4,4,6,70,70,126,0,68,68,68,1
26,98,98,98,0,64,64,64,96,96,96,126,0,12
6,74,74,106,106,106,106,0,126,66,66,98,9
8,98,98,0,126,66,66,98,98,98,126,0,126,6
6,126,96,96,96,96,0,126,66,66,98,98,110,
126,0,124,68,68,126,98,98,98,0,126,64,64
,126,6,6,126,0
 188 DATA 126,16,16,24,24,24,24,0,66,66,
66,98,98,98,126,0,98,98,98,98,36,36,60,0
,74,74,74,106,106,106,126,0,66,66,66,60,
98,98,98,0,66,66,66,126,24,24,24,0,126,2
,2,126,96,96,126,0
 190 FOR I=64384 TO 64463: READ A: POKE
I,A: NEXT I
 200 DATA 0,126,66,66,70,70,70,126,0,8,8
,8,24,24,24,24,0,126,66,2,126,96,96,126,
0,124,68,4,30,6,70,126,0,120,72,72,72,12
6,24,24,0,126,64,64,126,6,70,126,0,126,6
4,64,126,98,98,126,0,126,2,2,6,6,6,6,0,6
0,36,36,126,70,70,126,0,126,66,66,126,6,
6,6,0
 298 LET M$="                                "
 300 LET M$=M$+"         JJJBCCC
  "
 302 LET M$=M$+"         J      C C
  "
 304 LET M$=M$+"         J   B  C C
  "
 306 LET M$=M$+"         J BBB C C A    K K
  "
 308 LET M$=M$+"SSSSSJJJ    B CCCAAA   R R
  "
```

```
310 LET M$=M$+"S    S  R    B     A     R R
 "
312 LET M$=M$+"S    S       B  B A     R R
 "
314 LET M$=M$+"S    S RBB   BBBB A     HHH
 "
316 LET M$=M$+"S    S    B  B     A     H L
 "
318 LET M$=M$+"S    S    B  B     AHHHHH H
 "
320 LET M$=M$+"SSSSSSR B   B     A       H
 "
322 LET M$=M$+"  S       B  B     A       H
 "
324 LET M$=M$+"B B       B  B     AAAA    H
 "
326 LET M$=M$+"BBBBBBBBBBBBBBBA        H
 "
328 LET M$=M$+"B                  A      SH
H "
330 LET M$=M$+"B                  A      M
 "
332 LET M$=M$+"B        C    UUUU A      M
 "
334 LET M$=M$+"B        C    U  U F      M
 "
336 LET M$=M$+"B C   CCCUUUU   U FMMMMMMM
 "
338 LET M$=M$+"B C   C  U      U F
 "
340 LET M$=M$+"B C   C  U      UUFFFFFF
 "
342 LET M$=M$+"B CCCC  U              FF
 "
344 LET M$=M$+"B    C   U               F
 "
346 LET M$=M$+"B    C   U               F
 "
348 LET M$=M$+"B    C   U           I   F
 "
```

```
 350 LET M$=M$+"B    C    U           I    F
  "
 352 LET M$=M$+"B    C    URI         I    F
  "
 354 LET M$=M$+"B    C      IIII    IIFFFF
  "
 356 LET M$=M$+"B    C          I    I    F
  "
 358 LET M$=M$+"BCCCC          IIIII
  "
 360 LET M$=M$+"
  "
 800 FOR I=0 TO 7: READ A: POKE USR "Q"+
I,A: NEXT I: DATA 0,0,48,88,220,255,255,
255
 900 PRINT AT 10,4;" FINISHED - HIT A KE
Y    "
 901 PRINT AT 5,7;"FANTASTIC JOURNEY"
 903 IF INKEY$="" THEN  GO TO 903
 910 PRINT AT 18,0;"   HIT ~I~ FOR INSTR
UCTIONS"
 911 FOR I=1 TO 10: NEXT I
 912 LET S$=INKEY$: IF S$="" THEN  GO TO
 911
 920 IF S$="I" THEN  GO SUB 3000
1000 PAPER 5: INK 0: BORDER 1: CLS : LET
 D$="W": LET FL=100: LET R=505: LET AL=1
00: LET TL=500: LET NL=100: LET SC=100:
PRINT AT 21,0;"  LEFT FEMORAL VEIN."
1010 PRINT AT 1,10;"FANTASTIC JOURNEY";A
T 3,11;"TIME LEFT :-      500";AT 4,11;"N
EURONE LASER :- 100%";AT 5,11;"SUB CONDI
TION:-  100%";AT 6,11;"OXYGEN LEFT :-
100";AT 7,11;"FOOD LEFT :-      100"
1020 PRINT AT 19,12;"LOCATION"
1030 INK 3: PRINT AT 1,3;"abc";AT 2,3;"d
ef";AT 5,3; INK 2;"gh";AT 8,3;"ij";AT 10
,4; INK 3;"k";AT 11,3;"lm"
1040 INK 0: PLOT 8,160: DRAW 0,-90: DRAW
```

```
  58,0: PLOT 66,160: DRAW 0,-90
1050 PLOT 8,160: DRAW 15,0: PLOT 66,160:
 DRAW -18,0
1060 PLOT 8,132: DRAW 15,0: PLOT 66,132:
 DRAW -26,0
1070 PLOT 8,109: DRAW 15,0: PLOT 66,107:
 DRAW -26,0
1080 PLOT 66,90: DRAW -26,0
1090 PLOT 35,96: DRAW 0,7
1100 PLOT 95,104: DRAW 121,0: DRAW 0,-41
: DRAW -121,0: DRAW 0,41
1110 GO SUB 6000
1120 GO TO 7000
3000 CLS : PRINT AT 1,0;"        FANTASTI
C JOURNEY"
3002 PRINT AT 4,0;" THE FAMOUS DR HANS
 ZARKOV    HAS CONTRACTED A BRAIN TUMOUR
   YOUR MISSION IS TO SEEK OUT      THIS
TUMOUR AND DESTROY IT.        YOU WILL BE
 SHRUNK DOWN IN    SIZE AND INJECTED INT
O THE       DOCTOR'S BODY. YOUR SUBMARINE
   IS EQUIPPED WITH A NEURONE       LASER
, CAPABLE OF DESTROYING    THE TUMOUR. I
T WILL ONLY WORK   WHEN OVER 90% EFFICIE
NT. YOU    WILL ALSO HAVE TO WORK TO A
   TIME LIMIT AFTER WHICH YOU WILL BE EX
CRETED  FROM THE BODY.
                                  GOOD LUCK!
"
3010 PRINT AT 21,0;"   HIT ANY KEY TO CO
NTINUE "
3011 IF INKEY$="" THEN  GO TO 3011
3020 CLS : PRINT AT 1,0;"        FANTASTI
C JOURNEY"
3021 PRINT AT 4,0;"   YOU WILL ENCOUNTER
 MANY       DANGERS. TO CONTROL YOUR CRAF
T:-
        5- ROTATE LEFT
8- ROTATE RIGHT                  6- ROTAT
```

```
E 180'                          7- MOVE FORWARD
                                                TO
 THE LEFT OF THE SCREEN IS A BASIC MAP O
F THE BODY, BUT IT IS MORE COMPLICATED T
HAN THIS.  OXYGEN HAS TO BE USED IN VEIN
S  AS THERE IS A LACK OF IT IN THE VEINS
. FOOD STORES MAY BE        REPLENISHED W
HERE APPROPRIATE."
3030 PRINT AT 21,0;"   HIT ANY KEY TO PL
AY GAME"
3040 IF INKEY$<>"" THEN  RETURN
3050 GO TO 3040
6000 FOR I=9 TO 13: PRINT AT I,12;"
          ": NEXT I
6001 IF D$<>"N" THEN  GO TO 6100
6002 IF M$(R-1)<>" " THEN  INK 0: PLOT 9
5,96: DRAW 24,0: DRAW 0,-25: DRAW -24,0
6004 IF M$(R+1)<>" " THEN  INK 0: PLOT 2
16,96: DRAW -24,0: DRAW 0,-25: DRAW 24,0
6010 IF M$(R-1)=" " THEN  INK 0: PLOT 95
,104: DRAW 24,-8: DRAW 0,-24: DRAW -24,-
8
6020 IF M$(R+1)=" " THEN  INK 0: PLOT 21
6,104: DRAW -24,-8: DRAW 0,-24: DRAW 24,
-8
6030 IF M$(R-26)=" " THEN  INK 0: PLOT 1
19,96: DRAW 72,0: PLOT 119,71: DRAW 72,0
6040 RETURN
6100 IF D$<>"S" THEN  GO TO 6200
6102 IF M$(R+1)<>" " THEN  INK 0: PLOT 9
5,96: DRAW 24,0: DRAW 0,-25: DRAW -24,0
6104 IF M$(R-1)<>" " THEN  INK 0: PLOT 2
16,96: DRAW -24,0: DRAW 0,-25: DRAW 24,0
6110 IF M$(R+1)=" " THEN  INK 0: PLOT 95
,104: DRAW 24,-8: DRAW 0,-24: DRAW -24,-
8
6120 IF M$(R-1)=" " THEN  INK 0: PLOT 21
6,104: DRAW -24,-8: DRAW 0,-24: DRAW 24,
-8
```

```
6130 IF M$(R+26)=" " THEN  INK 0: PLOT 1
19,96: DRAW 72,0: PLOT 119,71: DRAW 72,0
6140 RETURN
6200 IF D$<>"W" THEN  GO TO 6300
6202 IF M$(R+26)<>" " THEN  INK 0: PLOT
95,96: DRAW 24,0: DRAW 0,-25: DRAW -24,0
6204 IF M$(R-26)<>" " THEN  INK 0: PLOT
216,96: DRAW -24,0: DRAW 0,-25: DRAW 24,
0
6210 IF M$(R+26)=" " THEN  INK 0: PLOT 9
5,104: DRAW 24,-8: DRAW 0,-24: DRAW -24,
-8
6220 IF M$(R-26)=" " THEN  INK 0: PLOT 2
16,104: DRAW -24,-8: DRAW 0,-24: DRAW 24
,-8
6230 IF M$(R-1)=" " THEN  INK 0: PLOT 11
9,96: DRAW 72,0: PLOT 119,71: DRAW 72,0
6240 RETURN
6300 IF M$(R-26)=" " THEN  INK 0: PLOT 9
5,104: DRAW 24,-8: DRAW 0,-24: DRAW -24,
-8
6302 IF M$(R-26)<>" " THEN  INK 0: PLOT
95,96: DRAW 24,0: DRAW 0,-25: DRAW -24,0
6304 IF M$(R+26)<>" " THEN  INK 0: PLOT
216,96: DRAW -24,0: DRAW 0,-25: DRAW 24,
0
6310 IF M$(R+26)=" " THEN  INK 0: PLOT 2
16,104: DRAW -24,-8: DRAW 0,-24: DRAW 24
,-8
6320 IF M$(R+1)=" " THEN  INK 0: PLOT 11
9,96: DRAW 72,0: PLOT 119,71: DRAW 72,0
6330 RETURN
7000 IF M$(R)<>"A" THEN  GO TO 7010
7004 PRINT AT 21,0;"              AORTA
          ": LET TL=TL-1: LET FL=FL-1: G
O SUB 9000
7006 GO SUB 6000: PRINT AT 16,0;"OXYGEN
LEVEL HIGH-PRESSURE HIGH":
7007 LET TL=TL-1
7008 GO TO 9500
```

```
7010 IF M$(R)<>"K" THEN  GO TO 7020
7011 PRINT AT 21,0;"              KIDNEY
          ": PRINT AT 16,0;"KIDNEYS ENTE
RED-YOU HAVE BEEN  EXCRETED FROM THE BO
DY.          GAME OVER-SCORE ";TL+NL+SC+F
L;"                                    ": PAU
SE 200: GO TO 9900
7020 IF M$(R)<>"R" THEN  GO TO 7030
7021 PRINT AT 16,0;"
          ": PRINT AT 21,0;"
RENAL ARTERY": LET TL=TL-1: LET FL=FL-1:
 GO SUB 9000: GO TO 9500
7030 IF M$(R)<>"M" THEN  GO TO 7040
7031 PRINT AT 16,0;"
          ": PRINT AT 21,0;"        MES
ENTERIC ARTERY        ": LET TL=TL-1: LE
T FL=FL-1: GO SUB 9000: GO TO 9500
7040 IF M$(R)<>"H" THEN  GO TO 7050
7041 PRINT AT 21,0;"      HEPATIC PORTAL
 VEIN       ": PRINT AT 16,0;"SUGAR LEVE
L HIGH-TAKE ON STORES?"
7042 LET S$=INKEY$: IF S$="" THEN  GO TO
 7042
7043 IF S$="Y" THEN  LET FL=100: LET TL=
TL-5
7044 LET TL=TL-1: LET AL=AL-2: GO SUB 90
00: GO TO 9500
7050 IF M$(R)<>"H" THEN  GO TO 7060
7051 PRINT AT 16,0;"
          ";AT 21,0;"        HEPATIC A
RTERY          ": LET TL=TL-1: LET FL=FL
-1: GO SUB 9000
7052 IF RND>.5 THEN  LET TL=TL-4: LET AL
=AL-INT (5*RND): LET FL=FL-INT (10*RND):
 PRINT AT 16,0;"ATTACKED BY LEUCOCYTES-D
AMAGED  ": LET SC=SC-INT (15*RND): LET N
L=NL-1: GO SUB 9000
7053 GO TO 9500
7060 IF M$(R)<>"C" THEN  GO TO 7070
```

```
7061 PRINT AT 21,0;"          CAROTID ART
ERY          ": PRINT AT 16,0;"
                          ": LET TL=TL-1: LE
T FL=FL-1: GO SUB 9000
7062 GO TO 9500
7070 IF M$(R)<>"J" THEN  GO TO 7080
7071 PRINT AT 16,0;"
             ";AT 21,0;"            JUGULAR
 VEIN           ": LET AL=AL-2: LET FL=FL
-1: LET TL=TL-1: GO SUB 9000: GO TO 9500
7080 IF M$(R)<>"I" THEN  GO TO 7090
7082 PRINT AT 16,0;"
             ";AT 21,0;"           LEFT LE
G VEIN          ": LET TL=TL-1: LET FL=FL
-1: LET AL=AL-1: GO SUB 9000: IF RND>.7
THEN  LET TL=TL-1: LET FL=FL-1: LET SC=S
C-INT (20*RND): PRINT AT 16,0;"ATTACKED
BY LEUCOCYTES-DAMAGED  ": GO SUB 9000: G
O TO 9500
7084 GO TO 9500
7090 IF M$(R)<>"F" THEN  GO TO 7095
7092 PRINT AT 16,0;"
             ";AT 21,0;"     LEFT FEMORAL
 ARTERY         ": LET TL=TL-1: LET FL=FL
-1: GO SUB 9000: GO TO 9500
7093 GO TO 9500
7095 IF M$(R)<>"C" THEN  GO TO 8000
7097 PRINT AT 16,0;"
             ";AT 21,0;"      RIGHT FEMOR
AL VEIN         ": LET TL=TL-1: LET FL=FL
-1: LET AL=AL-2: GO SUB 9000
7098 IF RND>.8 THEN  LET FL=FL-2: LET AL
=AL-1: LET NL=NL-1: LET SC=SC-INT (10*RN
D): GO SUB 9000: PRINT AT 16,0;"ATTACKED
 BY LEUCOCYTES-DAMAGED": GO TO 9500
7099 GO TO 9500
8000 IF M$(R)<>"U" THEN  GO TO 8005
8002>PRINT AT 16,0;"
             ";AT 21,0;"    RIGHT FEMORAL
 ARTERY         ":LET TL=TL-1:LET FL=FL-1
:GO SUB 9000:IF RND>.8THEN PRINT AT 16,0
;"ATTACKED BY LEUCOCYTES-DAMAGED  ":LET
```

```
TL=TL-5:LET SC=SC-INT (20*RND)
8004 GO TO 9500
8005 IF M$(R)<>"B" THEN  GO TO 8010
8006 PRINT AT 16,0;"
            ";AT 21,0;"        INFERIOR V
ENA CAVA        ": LET TL=TL-1: LET AL=AL
-1: LET FL=FL=1: GO SUB 9000: IF RND>.9
THEN  PRINT AT 16,0;"CRUSHED BY A VALVE
 GAME OVER   -   SCORE ";TL+SC+FL+NL: PA
USE 200: GO TO 9900
8008 GO TO 9500
8010 IF M$(R)<>"S" THEN  GO TO 8020
8012 PRINT AT 16,0;"
            ";AT 21,0;"        SUPERIOR V
ENA CAVA        ": LET TL=TL-1: LET AL=AL
-1: LET FL=FL=1: GO SUB 9000: IF RND>.9
THEN  PRINT AT 16,0;"CRUSHED BY A VALVE
- GAME OVER       SCORE ";TL+SC+FL+NL: PA
USE 200: GO TO 9900
8014 GO TO 9500
8020 IF M$(R)<>"S" THEN  GO TO 8030
8022 PRINT AT 21,0;"             STOMACH
            "
8023 IF SC<40 THEN  PRINT AT 16,0;"SHIEL
D DISSOLVED BY ACID ----    GAME OVER - S
CORE ";TL+SC+NL+FL: PAUSE 200: GO TO 990
0
8024 PRINT AT 16,0;"SHIELDS HOLDING-DAMA
GE TAKEN    ": LET SC=SC-INT (10*RND)
8025 LET TL=TL-1: LET FL=FL-1: LET AL=AL
-1: GO SUB 9000: GO TO 9500
8030 IF M$(R)<>"L" THEN  GO TO 8040
8032 LET TL=TL-1: LET AL=AL-1: LET FL=FL
-1: PRINT AT 21,0;"               LIVER
            ";AT 16,0;"SUGAR LEVEL HIGH-T
AKE ON STORES?"
8033 LET S$=INKEY$: IF S$="" THEN  GO TO
 8033
8034 IF S$="Y" THEN  LET TL=TL-5: LET FL
=100: GO SUB 9000: PRINT AT 16,0;"
                          ": GO TO 9500
```

```
8035 PRINT AT 16,0;"
            ": GO SUB 9000: GO TO 9500
8040 IF M$(R)<>"R" THEN  GO TO 8050
8042 PRINT AT 21,0;"                HEART
            ";AT 16,0;"PASSING THORUGH R
IGHT ATRIUM     ": IF RND>.6 THEN  PAUSE
200: PRINT AT 16,0;"CRUSHED BY TRICUSPID
 VALVE       GAME OVER - SCORE ";SC+NL+FL
+TL: PAUSE 200: GO TO 9900
8043 PRINT AT 16,0;"PASSING THROUGH RIGH
T VENTRICLE ": PAUSE 200: IF RND>.6 THEN
  PRINT AT 16,0;"CRUSHED BY CARDIAC MUSC
LE        GAME OVER - SCORE ";TL+SC+NL+FL
: PAUSE 200: GO TO 9900
8044 PRINT AT 16,0;"PASSING DOWN PULMONA
RY ARTERY   AND INTO LUNG -----
    ": PAUSE 200: PRINT AT 16,0;"PASSING
 DOWN PULMONARY VEIN AND INTO HEART AT
LEFT ATRIUM --   ": IF RND>.6 THEN  PAUS
E 200: PRINT AT 16,0;"CRUSHED BY BICUSPI
D VALVE        GAME OVER - SCORE ";TL+FL+
NL+SC: PAUSE 200: GO TO 9900
8046 PAUSE 200: PRINT AT 16,0;"PASSING T
HROUGH LEFT VENTICLE   ": PAUSE 200: IF
RND>.7 THEN  PRINT AT 16,0;"CRUSHED BY C
ARDIAC MUSCLE        GAME OVER - SCORE ";
FL+SC+NL+TL: PAUSE 200: GO TO 9900
8048 PRINT AT 16,0;"PASSING OUT OF AORTA
 ------      ": LET TL=TL+5: LET AL=100:
LET FL=FL-3: GO TO 9000
8049 LET R=485: GO SUB 9000: GO TO 9500
8050 FOR I=9 TO 13: PRINT AT I,12;"
          ": NEXT I
8052 PRINT AT 10,13;"abc";AT 11,13;"def"
8054 FOR I=0 TO 7: READ A: POKE USR "N"+
I,A: NEXT I
8055 FOR I=0 TO 7: READ A: POKE USR "O"+
I,A: NEXT I
8056 DATA 0,31,50,205,50,31,0,0,0,0,193,
58,78,58,193,128,192
8057 PRINT AT 10,23;"no"
```

```
8058 INK 4: FOR I=183 TO 128 STEP -1: BE
EP .01,0
8059 PLOT I,92: NEXT I
8060 FOR I=0 TO 5: INK I: PAUSE 5: PRINT
 AT 10,13;"abc";AT 11,13;"def": NEXT I
8061 INK 5: FOR I=128 TO 183: PLOT I,92:
 NEXT I
8062 INK 0: PRINT AT 16,0;"TUMOUR DESTRO
YED - WELL DONE ¤¤¤SCORE - ";1000+NL+SC+
FL+AL+TL: PAUSE 500: GO TO 9990
9000 PRINT AT 3,28;TL;" ";AT 4,28;NL;"x"
;AT 5,28;SC;"x";AT 6,28;AL;" ";AT 7,28;F
L;" "
9010 IF NL<100 THEN  PRINT AT 4,31;" "
9015 IF SC<100 THEN  PRINT AT 5,31;" "
9016 IF SC<10 THEN  PRINT AT 5,30;" "
9017 IF SC<0 THEN  PRINT AT 5,28;"0x  "
9018 IF TL<100 THEN  PRINT AT 3,28;TL;"
": IF TL<10 THEN  PRINT AT 3,28;TL;"   "
9020 IF AL<100 THEN  PRINT AT 6,28;AL;"
": IF AL<10 THEN  PRINT AT 6,28;AL;"   "
9021 IF FL<100 THEN  PRINT AT 7,28;FL;"
": IF FL<10 THEN  PRINT AT 7,28;FL;"   "
9022 IF TL<=0 THEN  PRINT AT 16,0;"OUT O
F TIME - SCORE ";TL+NL+SC+FL: PAUSE 200:
 GO TO 9990
9023 IF NL<=89 THEN  PRINT AT 16,0;"NEUR
ONE LASER INEFFECTIVE        GAME OVER -
SCORE ";TL+NL+FL+SC: PAUSE 200: GO TO 99
90
9024 IF TL<=0 OR NL<=89 OR SC<=0 OR AL<=
0 OR FL<=0 THEN  GO TO 9900
9025 GO SUB 6000
9030 RETURN
9500 LET S$=INKEY$: IF S$="" THEN  GO TO
 9500
9510 IF D$<>"N" THEN  GO TO 9520
9511 IF S$="5" THEN  LET D$="W"
9512 IF S$="8" THEN  LET D$="E"
9513 IF S$="6" THEN  LET D$="S"
9514 IF S$="7" AND M$(R)="A" THEN  PRINT
```

```
 AT 16,0;"   BLOOD PRESSURE TOO HIGH
   ": GO TO 7000
9515 IF S$="Q" THEN  PRINT AT 15,0;"GAME
 OVER - SCORE ";NL+SC+FL+TL: PAUSE 200:
GO TO 9900
9516 IF S$="7" AND M$(R-26)<>" " THEN  L
ET R=R-26: GO TO 7000
9517 GO TO 7000
9520 IF D$<>"S" THEN  GO TO 9530
9521 IF S$="5" THEN  LET D$="E"
9522 IF S$="8" THEN  LET D$="W"
9523 IF S$="Q" THEN  PRINT AT 16,0;"GAME
 OVER-SCORE ";NL+SC+TL+FL: PAUSE 200: GO
 TO 9900
9524 IF S$="6" THEN  LET D$="N"
9525 IF S$="7" AND M$(R+26)<>" " THEN  L
ET R=R+26
9526 GO TO 7000
9530 IF D$<>"W" THEN  GO TO 9540
9531 IF S$="5" THEN  LET D$="S"
9532 IF S$="8" THEN  LET D$="N"
9533 IF S$="6" THEN  LET D$="E"
9534 IF S$="7" AND M$(R-1)<>" " THEN  LE
T R=R-1
9535 IF S$="Q" THEN  PRINT AT 16,0;"GAME
 OVER-SCORE ";NL+SC+TL+FL: PAUSE 200: GO
 TO 9900
9536 GO TO 7000
9540 REM EAST
9541 IF S$="5" THEN  LET D$="N"
9542 IF S$="8" THEN  LET D$="S"
9543 IF S$="6" THEN  LET D$="W"
9544 IF S$="7" AND M$(R+1)<>" " THEN  LE
T R=R+1
9545 IF S$="Q" THEN  PRINT AT 16,0;"GAME
 OVER-SCORE ";NL+SC+TL+FL: PAUSE 200: GO
 TO 9900
9546 GO TO 7000
9910 FOR I=20 TO 0 STEP -1
9920 PRINT AT I,0;"qqqqqqqqqqqqqqqqqqqqq
```

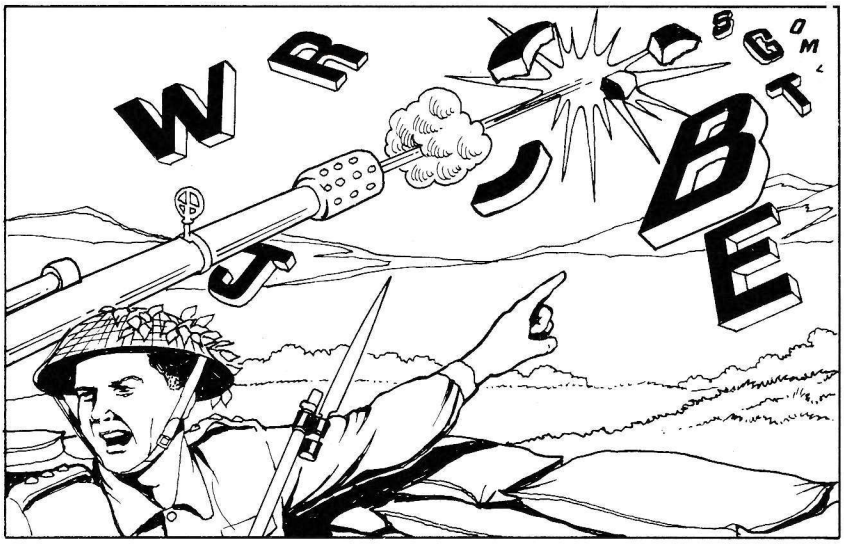
```
qqqqqqqqqqq"; INVERSE 1;"

9924 BEEP 0.05,I
9925 INVERSE 0
9930 NEXT I
9990 PAUSE 200: INK 0: PAPER 5: CLS : PR
INT AT 16,0;"PLAY AGAIN ? ( Y OR N )


9991 LET S$=INKEY$: IF S$="" THEN  GO TO
 9991
9992 IF S$="N" THEN  NEW
9993 IF S$="Y" THEN  RUN
```

Chapter 2

Type Attack



Description

This game is designed for the younger user or someone who has just acquired his first computer and is trying to find his way around the keyboard and familiarise himself with the position of the various letters.

As the title suggests you will be attacked by various letters of the alphabet which will move down the screen towards your base. These letters can be despatched by typing the corresponding key. If your knowledge of the keyboard is good and your reflexes ultrafast then you might just manage to score a hit in the cyan bonus area.

```
  30 PRINT AT 3,0;"  YESTERDAY, THE WORD
 PROCESSORSOF THE WORLD REBELLED,
   THREATENING THE VERY EXISTANCE  OF MA
NKIND !!!
                        LETTERS RAINING FROM
THE SKY !!!FORTUNATELY, THE MEGA-BEVAWAT
T  LASER TURRETS HAVE BEEN SET UP  SO TH
AT BY TOUCHING THE KEY     CORRESPONDING
 TO THE LETTER     FALLING WILL CAUSE TH
AT LETTER  TO RETREAT. DESTROYING A LETT
ER IN THE IONOSPHERE GIVES A BONUS!

                                    GOOD LUCK!
"
  40 PRINT AT 2,0;" ";AT 2,31;" ";AT 18,
31;" "
  50 INPUT "START LEVEL-(0=FAST-9=SLOW)
";SP
  60 IF SP<0 OR SP>9 OR SP<>INT SP THEN
 GO TO 50
1000 FOR A=1 TO 10: CLS
1010 BORDER 7: PAPER 7: INK 0
1011 PRINT AT 0,0;"LIVES - ";LIVES;" SCO
RE- ";S$;" HI- ";H$
1020 PRINT AT 21,0; INVERSE 1;"
                          "; INVERSE 0
1021 PRINT AT 21,12; INVERSE 1;"LEVEL ";
SP
1022 PRINT AT 5,0; PAPER 5;"

                "

1025 PLOT 0,8: DRAW 8,15: DRAW 8,-15: PL
OT 255,8: DRAW -8,15: DRAW -8,-15
1030 LET L=INT (RND*26)
1040 LET L$=CHR$ (65+L)
1050 LET X=L+3
1060 FOR Y=2 TO 20
1065 PAPER (ATTR (Y-1,X)/8): PRINT AT Y-
1,X;" "
```

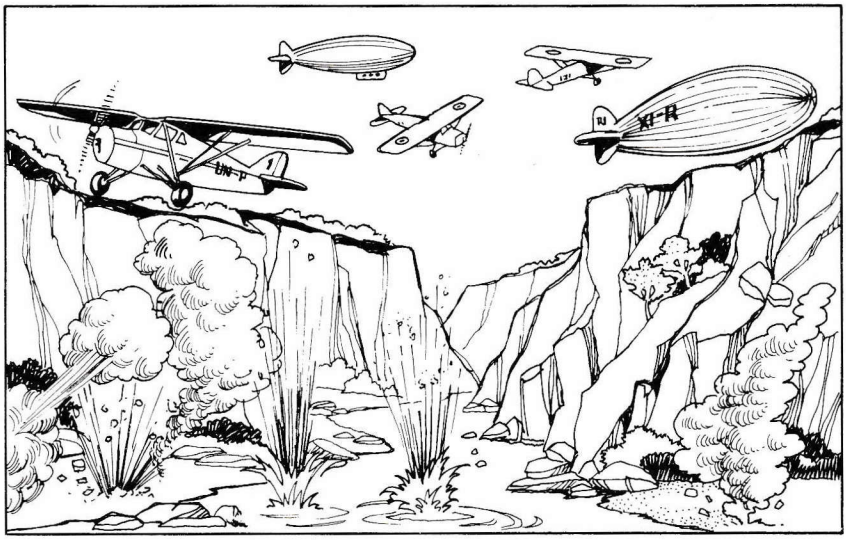
```
1070 PAPER (ATTR (Y,X)/8): PRINT AT Y,X;
L$
1080 FOR I=1 TO SP: NEXT I
1090 LET Q$=INKEY$: IF Q$="" THEN  GO TO
 1150
1100 IF Q$=L$ THEN  GO TO 2000
1150 NEXT Y
1151 LET Y=21: PRINT AT 20,X;" "
1200 FOR G=40 TO 25 STEP -1: PRINT AT 21
,X; INK 2;CHR$ 134: BEEP .05,G: PRINT AT
 21,X;CHR$ 137: BEEP .05,G-1: PRINT AT 2
1,X;CHR$ 134: BEEP .05,G-2: PRINT AT 21,
X;CHR$ 137: BEEP .05,G-3: NEXT G
1210 INK 0: LET LIVES=LIVES-1: PRINT AT
0,8;LIVES: IF LIVES=0 THEN  GO TO 9000
1220 CLS : NEXT A
1230 IF SP<>0 THEN  LET SP=SP-1
1240 GO TO 1000
2000 INK 2: IF L>13 THEN  GO TO 2500
2010 PLOT 8,24: DRAW (8*(X-1)),((18-Y)*8
)+4
2020 INK 0: GO TO 2600
2500 PLOT 247,24: DRAW -(8*(30-X)),((18-
Y)*8)+4
2600 IF Y>=5 AND Y<=6 THEN  LET SCORE=SC
ORE+1000: FOR I=1 TO 50: BEEP .001,20: B
EEP .001,30: NEXT I: PAPER 7: INK 0: GO
TO 2621
2601 BEEP .1,10
2605 INK 7: PLOT 8,24: DRAW (8*(X-1)),((
18-Y)*8)+4: PLOT 247,24: DRAW -(8*(30-X)
),((18-Y)*8)+4
2610 INK 0: PLOT 0,8: DRAW 8,15: DRAW 8,
-15: PLOT 255,8: DRAW -8,15: DRAW -8,-15
2613 INK 0: FOR G=Y TO 2 STEP -1: IF G<=
4 OR G>=7 THEN  PAPER 7: PRINT AT G,X;"
": GO TO 2615
2614 PAPER 5: PRINT AT G,X;" "
2615 IF G-1<=4 OR G-1>=7 THEN  PAPER 7:
```

```
PRINT AT G-1,X;L$: GO TO 2617
2616 PAPER 5: PRINT AT G-1,X;L$: PAPER 7
2617 BEEP .01,21-G: NEXT G
2620 PRINT AT 1,X;" "
2621 LET SCORE=SCORE+(21-Y): LET S$="000
00"+STR$ SCORE: LET S$=S$(LEN S$-4 TO LE
N S$)
2625 IF VAL S$>VAL H$ THEN  LET H$=S$
2630 CLS : NEXT A
2640 IF SP<>0 THEN  LET SP=SP-1
2650 GO TO 1000
9000 PRINT AT 10,10; INK 0; FLASH 1;"GAM
E OVER"; FLASH 0
9010 IF SCORE>T(10) THEN  GO TO 9100
9020 PRINT AT 12,8;"PLAY AGAIN?"
9030 IF INKEY$="Y" THEN  LET LIVES=3: LE
T SCORE=0: LET S$="00000": GO TO 1000
9040 IF INKEY$="N" THEN  STOP
9050 GO TO 9030
9100 FOR J=1 TO 300: NEXT J
9110 CLS
9120 FLASH 1: PRINT AT 10,10;"CONGRATULA
TIONS"
9130 PRINT AT 12,7;"YOU ARE ON THE TABLE
"
9140 FLASH 0
9150 FOR I=1 TO 40
9160 BEEP .01,I
9170 NEXT I
9180 INPUT "ENTER YOUR NAME=";N$
9190 CLS
9200 PRINT TAB (10);"HIGHEST SCORES"
9210 PRINT TAB (10);"======= ======"
9220 PRINT
9230 FOR I=1 TO 10
9240 IF SCORE>T(I) THEN  GO TO 9270
9250 NEXT I
9260 GO TO 9330
9270 FOR J=9 TO I STEP -1
```

```
9280 LET T(J+1)=T(J)
9290 LET T$(J+1)=T$(J)
9300 NEXT J
9310 LET T(I)=SCORE
9320 LET T$(I)=N$
9330 FOR I=1 TO 10
9340 PRINT T$(I),T(I)
9350 NEXT I
9400 PRINT : PRINT
9410 PRINT TAB (8);"PLAY AGAIN?"
9420 GO TO 9030
```

Chapter 3

Canyon Bomber



Description

The object of the game is to fly your slow-moving balloons and faster single-wing bombers over the great canyon, trying to destroy as much of your enemy's supplies as possible. Unfortunately, the bombs must be dropped during a single pass over the canyon since they are very unstable and can detonate onboard if they are not jettisoned within a given period of time. You have five craft in all, but the enemy has a lot of supplies. Good luck, good hunting and bombs away.

Hints on entry

The game is relatively straightforward to enter provided that the usual care is taken over the user-defined graphics. These are represented by lower case letters as opposed to the upper case letters used for the remainder of the program. Care should also be taken with the data lines (9100–9140 and 9260–9330), since an incorrect entry in this section will cause errors in the graphical display.

Techniques

One facility often included in many commercial packages is a high score table which displays on the screen the ten best scores obtained in the game. Such a facility has been included in this and several other games in the book. The methods employed to obtain such a table are described below.

1) Two arrays are set up—one for names and the other for the corresponding scores—and the starting values to be used at the beginning of the game are assigned.

```
10 REM DEFINE HIGH SCORE TABLE
20 DIM H$(10,10): DIM H(10)
30 FOR I=1 TO 10
40 LET H$(I)="FRED"
50 LET H(I)=100
60 NEXT I
```

2) During the game the score, represented by some variable S, will be updated. Upon completion of the game, the program should branch to a subroutine which checks the player's score against the lowest score on the table and then takes the appropriate action.

```
9000 REM HIGH SCORE TABLE: CLS
9010 IF S<H(10) THEN  GO TO 9150
9020 PRINT "CONGRATULATIONS"
9030 INPUT "ENTER YOUR NAME...";A$
9040 IF LEN (A$)>10 THEN  GO TO 9030
9050 CLS
```

```
9060 FOR I=1 TO 10
9070 IF S>H(I) THEN  GO TO 9090
9080 NEXT I
9090 FOR J=9 TO I STEP -1
9100 LET H(J+1)=H(J)
9110 LET H$(J+1)=H$(J)
9120 NEXT J
9130 LET H(I)=S
9140 LET H$(I)=A$
9150 PRINT TAB (10);"HIGHEST SCORES"
9160 PRINT TAB (10);"======= ======"
9170 PRINT
9180 FOR I=1 TO 10
9190 PRINT H$(I),H(I)
9200 NEXT I
9210 RETURN
```

Explanation

9010	If score is not high enough to register on the table then jump to printing section.
9020–9040	Enter and check name
9060–9080	Print headings.
9090–9110	Look for point of insertion on table.
9120–9170	Insert name and move all other records down one place.
9180–9210	Print updated table and return to main part of program.

This routine could be used in all games programs employing the variable S for the score.

Instructions

Load the game from tape or microdrive by typing:

```
LOAD""
```

or

```
LOAD *"M";1;"CANYON"
```

When loaded the game will run automatically. A brief description will be displayed on the screen after which there will be a short delay while the computer initialises the screen and the variables.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED IN LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
  1 LET S=0: LET L=5: LET HS=0
  2 PAPER 7: POKE 23658,8
  3 FOR I=1 TO 80
  4 READ A
  5 POKE USR "A"+I-1,A
  6 NEXT I
  7 DATA 32,55,63,63,55,33,0,0,0,254,25
5,255,254,16,224,224,0,127,255,255,127,8
,7,7,4,236,252,252,236,132,0,0
  8 DATA 96,112,127,127,127,2,1,1,0,49,
253,255,253,65,128,128,0,140,191,255,191
,130,1,1,6,14,254,254,254,64,128,128
  9 DATA 128,192,254,255,254,192,128,0,
1,3,127,255,127,3,1,0
 10 CLS
 11 DIM S(28,12)
 12 DIM H$(10,10)
 13 DIM H(10)
 14 FOR I=1 TO 10: LET H(I)=0: LET H$(I
)="CESIL": NEXT I
 15 GO SUB 9900
 20 GO SUB 9000
 30 GO SUB 9200
 40 GO SUB 8000
 50 PAPER 7: INK 0
 60 PRINT AT 1,3;"SCORE=";S
 70 PRINT AT 1,22;"LIVES=";L
```

```
  80 PRINT AT 2,10;"HIGH SCORE=";HS
  90 PLOT 0,175: DRAW 255,0: DRAW 0,-30:
 DRAW -255,0: DRAW 0,30
 100 LET R=RND
 101 LET D=0
 102 LET S1=S
 105 PAPER 5
 110 IF R>.75 THEN  GO SUB 5000: GO TO 1
70
 120 IF R>.5 THEN  GO SUB 5100: GO TO 17
0
 130 IF R>.25 THEN  GO SUB 5200: GO TO 1
70
 140 GO SUB 5300
 170 INK 0: PAPER 7
 180 PRINT AT 1,9;S
 181 IF S=S1 THEN  LET L=L-1
 182 IF S/335=INT (S/335) THEN  GO TO 30
 183 PRINT AT 1,28;L
 184 IF L=0 THEN  FLASH 1: PRINT AT 7,10
;"GAME OVER": FLASH 0: PAUSE 100: GO TO
7000
 190 GO TO 100
 999 STOP
5000 FOR I=0 TO 31
5010 PRINT AT 5,I;"ab"
5020 IF INKEY$="0" AND D=0 THEN  GO TO 5
050
5025 PAUSE 1
5030 PRINT AT 5,I;"  "
5035 NEXT I
5040 RETURN
5050 FOR J=6 TO 20
5055 IF ATTR (J,I)=56 THEN  GO TO 5071
5060 PRINT AT J,I;"i"
5065 PRINT AT J,I;" "
5070 NEXT J
5071 IF I<3 OR I=31 THEN  GO TO 5075
5072 FOR K=1 TO 12
```

```
5073 LET S=S+S(I-2,K): LET S(I-2,K)=0
5074 NEXT K
5075 LET D=1
5080 GO TO 5025
5100 FOR I=31 TO 1 STEP -1
5110 PRINT AT 5,I;"cd"
5120 IF INKEY$="0" AND D=0 THEN  GO TO 5
150
5125 PAUSE 1
5130 PRINT AT 5,I;"  "
5135 NEXT I
5140 RETURN
5150 FOR J=6 TO 20
5155 IF ATTR (J,I)=56 THEN  GO TO 5171
5160 PRINT AT J,I;"j"
5165 PRINT AT J,I;" "
5170 NEXT J
5171 IF I<3 OR I=31 THEN  GO TO 5175
5172 FOR K=1 TO 12
5173 LET S=S+S(I-2,K): LET S(I-2,K)=0
5174 NEXT K
5175 LET D=1
5180 GO TO 5125
5200 FOR I=0 TO 31
5210 PRINT AT 5,I;"ef"
5220 IF INKEY$="0" AND D=0 THEN  GO TO 5
250
5230 PRINT AT 5,I;"  "
5235 NEXT I
5240 RETURN
5250 FOR J=6 TO 20
5255 IF ATTR (J,I)=56 THEN  GO TO 5271
5260 PRINT AT J,I;"i"
5265 PRINT AT J,I;" "
5270 NEXT J
5271 IF I<3 OR I=31 THEN  GO TO 5275
5272 FOR K=1 TO 12
5273 LET S=S+S(I-2,K): LET S(I-2,K)=0
5274 NEXT K
```

```
5275 LET D=1
5280 GO TO 5225
5300 FOR I=31 TO 1 STEP -1
5310 PRINT AT 5,I;"gh"
5320 IF INKEY$="0" AND D=0 THEN  GO TO 5
350
5330 PRINT AT 5,I;"  "
5335 NEXT I
5340 RETURN
5350 FOR J=6 TO 20
5355 IF ATTR (J,I)=56 THEN  GO TO 5371
5360 PRINT AT J,I;"J"
5365 PRINT AT J,I;" "
5370 NEXT J
5371 IF I<3 OR I=31 THEN  GO TO 5375
5372 FOR K=1 TO 12
5373 LET S=S+S(I-2,K): LET S(I-2,K)=0
5374 NEXT K
5375 LET D=1
5380 GO TO 5325
7000 REM *HIGH SCORE TABLE*
7005 LET HS=S
7010 CLS
7011 FLASH 1: PRINT AT 10,10;"CONGRATULA
TIONS"
7012 PRINT AT 12,7;"YOU ARE ON THE TABLE
"
7013 FLASH 0
7014 FOR I=1 TO 40
7015 BEEP .01,I
7016 NEXT I
7017 INPUT "ENTER YOU NAME=";N$
7018 CLS
7020 PRINT TAB (10);"HIGHEST SCORES"
7030 PRINT TAB (10);"======= ======="
7040 PRINT
7050 FOR I=1 TO 10
7060 IF HS>H(I) THEN  GO TO 7100
7070 NEXT I
7080 GO TO 7140
```

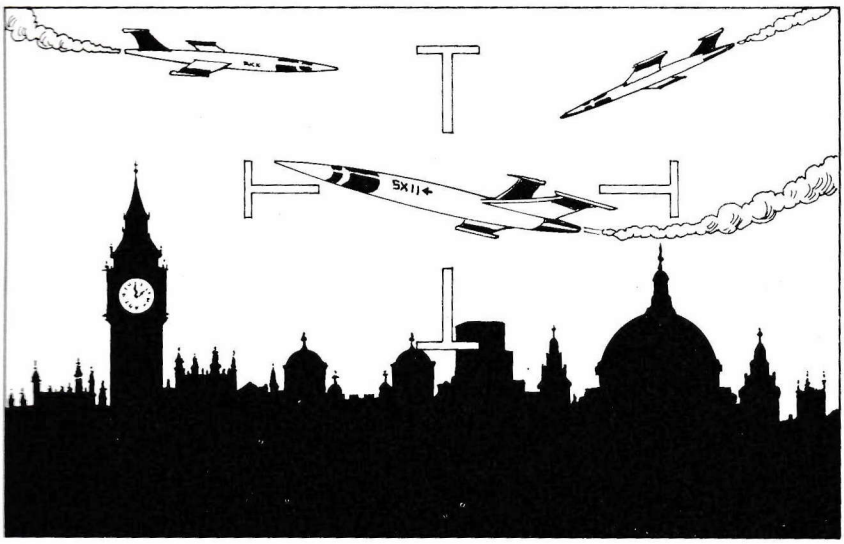
```
7100 FOR J=9 TO I STEP -1
7110 LET H(J+1)=H(J)
7115 LET H$(J+1)=H$(J)
7120 NEXT J
7130 LET H(I)=HS
7135 LET H$(I)=N$
7140 FOR I=1 TO 10
7150 PRINT H$(I),H(I)
7160 NEXT I
7170 INPUT "PRESS ENTER TO CONTINUE.";Z$
7171 LET L=5: LET S=0: LET HS=H(1)
7180 INPUT "DO YOU WANT ANOTHER GAME=";A
$
7190 IF A$(1)="Y" THEN  GO TO 15
7200 STOP
8000 REM *SCORE-BLOCKS*
8001 PAPER 5
8010 FOR X=1 TO 28
8020 FOR Y=1 TO 12
8025 IF S(X,Y)=0 THEN  GO TO 8040
8030 PRINT AT Y+8,X+2;CHR$ (48+S(X,Y))
8040 NEXT Y
8050 NEXT X
8060 RETURN
9000 REM *SCREEN*
9001 RESTORE 9100
9005 LET X=0: LET Y=4
9010 READ A,B
9020 INK B
9030 FOR J=1 TO A
9040 PRINT AT Y,X;CHR$ 143
9045 LET X=X+1
9050 NEXT J
9060 IF X=32 THEN  LET Y=Y+1: LET X=0
9070 IF Y<>22 THEN  GO TO 9010
9075 PAPER 5
9090 RETURN
9100 DATA 32,5,32,5,32,5,32,5
9110 DATA 1,0,31,5,2,0,30,5,4,0,28,5
9120 DATA 4,0,27,5,1,0,5,0,26,5,1,0,5,0,
25,5,2,0
9130 DATA 5,0,24,5,3,0,5,0,24,5,3,0,6,0,
```

```
22,5,4,0,6,0,21,5,5,0
9140 DATA 6,0,20,5,6,0,6,0,10,5,2,0,8,5,
6,0,6,0,10,5,3,0,7,5,6,0,32,0
9200 RESTORE 9260
9205 FOR Y=1 TO 12
9210 FOR X=1 TO 28
9220 READ A
9221 IF A=0 THEN  READ B
9222 IF A<>0 THEN  LET S(X,Y)=A: GO TO 9
230
9223 FOR I=1 TO B
9224 LET S(X,Y)=0
9225 LET X=X+1: IF X=29 THEN  GO TO 9240
9226 NEXT I
9227 IF X<>29 THEN  GO TO 9220
9230 NEXT X
9240 NEXT Y
9250 RETURN
9260 DATA 9,0,27,0,1,6,0,26,0,1,9,0,25,9
9270 DATA 0,2,4,0,23,8,9,0,2,3,0,22,7,9,
0,1
9280 DATA 0,2,1,1,0,20,6,9,0,2,0,2,9,3,1
,2,2,1,0,11,1,2,2,3,4,5,9,0,2
9290 DATA 0,3,2,1,2,3,1,0,10,4,2,1,3,1,4
,7,0,3
9300 DATA 0,3,3,1,3,1,1,4,6,3,0,6,3,1,1,
1,2,3,1,0,4
9310 DATA 0,3,1,1,1,2,2,2,3,4,2,4,5,4,7,
1,2,2,3,1,4,1,0,5
9320 DATA 0,3,2,4,2,3,2,4,1,1,1,6,0,2,3,
2,1,1,2,1,1,3,0,5
9330 DATA 0,3,9,1,1,2,2,2,3,1,2,7,0,3,4,
1,1,2,3,1,2,0,5
9900 REM *INTRODUCTION*
9910 CLS
9920 PRINT TAB (10);"CANYON BOMBER"
9930 PRINT TAB (10);"====== ======"
9940 PRINT
9942 PRINT "THE OBJECT OF THE GAME IS TO
"
```

```
9944 PRINT "SCORE AS MANY POINTS AS YOU
CAN"
9946 PRINT "BY DROPPING BOMBS INTO THE "
9948 PRINT "CANYON.(KEY-0)"
9950 PRINT "THE BOMBS ARE DROPPED, FROM
THE"
9952 PRINT "BALLOONS OR THE DIVE BOMBERS
 "
9954 PRINT "AS SHOWN BELOW."
9956 FOR I=1 TO 20
9957 PRINT AT 12,I;"ab"
9958 PAUSE 2
9959 IF I<>20 THEN  PRINT AT 12,I;"  "
9960 NEXT I
9961 PRINT AT 12,6;"BALLOONS"
9962 FOR I=1 TO 20
9963 PRINT AT 14,I;"ef"
9964 PAUSE 2
9965 IF I<>20 THEN  PRINT AT 14,I;"  "
9966 NEXT I
9967 PRINT AT 14,6;"DIVE BOMBERS"
9968 PRINT '''
9970 PRINT "YOU HAVE 5 LIVES WITH WHICH
TO"
9972 PRINT "COMPLETE YOUR MISSION, BUT T
HESE"
9974 PRINT "CAN BE LOST BY HITTING THE "
9976 PRINT "WALLS OR MISSING THE TARGET.
"
9978 INPUT "PRESS ENTER TO CONTINUE.";Z$
9979 CLS
9980 RETURN
```

Chapter 4

Missile Attack



Description

As commander-in-chief of the 1st Anti-Missile Battalion, it is your job to defend our great city from total destruction by the enemy's megaton super missiles. To protect the city, you have at your disposal two auto-ranging bevawatt lasers which you control from the underground defence computer situated beneath the centre of the city.

The screen of the defence computer shows the city, the laser cannon and the track of the enemy missiles as they descend, locked onto their target. For how long can you concentrate because the enemy never give in and the attack waves become more and more ferocious?

Hints on entry

As with the previous program (Canyon Bomber), Missile Attack should be straightforward to enter, the only problems being those relating to the user-defined graphics and the number of spaces (32) between the quotes in line 1. When the program has been entered, it should be saved on tape or microdrive by typing:

SAVE "MISSILE" LINE 1

or

SAVE *"M";1;"MISSILE" LINE 1

Techniques

Missile Attack has been designed to be totally controlled by a joystick connected to the Spectrum via the Sinclair Interface 2. The joystick is used to control movement, to fire the lasers and to enter a name should a high score be obtained. This is achieved by using the joystick to move the cursor along a screen menu and by using the fire button to enter a letter.

If you own an Interface 2, then the use of a joystick routine can enhance your program as well as save on the wear and tear of the keyboard caused by excessive use of one or more keys.

The most efficient way of using the joystick is to use the IN command to check for a particular key or combination of keys. A typical routine for testing the position of the joystick is shown below:

```
9000 REM JOYSTICK ROUTINE
9010 LET FIRE=0: LET HM=0: LET VM=0
9020 LET A=255-IN 61438
9030 IF A=0 THEN  RETURN
9040 IF A>127 THEN  LET A=A-128
9050 IF A>63 THEN  LET A=A-64
9060 IF A>31 THEN  LET A=A-32
9070 IF A>15 THEN  LET A=A-16: LET HM=-1
9080 IF A>7 THEN  LET A=A-8: LET HM=1
9090 IF A>3 THEN  LET A=A-4: LET VM=1
```

```
9100 IF A>1 THEN  LET A=A-2: LET VM=-1
9110 IF A=1 THEN  LET FIRE=1
9120 RETURN
```

When the computer returns from the subroutine to the main program, the horizontal movement and the vertical movement of the cursor are shown in the variables HM and VM, respectively, and if the fire button has been pressed, the variable FIRE will contain the number 1.

Instructions

Load the game from tape or microdrive by typing:

LOAD""

or

LOAD *"M";1;"MISSILE"

When loaded the game will run automatically, and the city, your laser guns and the surrounding district will be displayed on the screen. The enemy missiles are indicated by their tracks on the screen. When the attack starts you must use the joystick to intercept them by placing the missile in the centre of your sight. Each attack wave consists of ten missiles which must be destroyed before moving to the next level.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED IN LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
1 LET S$="
 ": LET T$=S$
2 FOR I=1 TO 19
3 LET S$=S$+T$: NEXT I
4 LET FIRE=0: LET QW=0
```

```
   5 DIM S(5): DIM G(5): DIM Y(5): DIM R
(5)
   6 POKE 23658,8
   9 GO SUB 9900
  10 CLS
  11 LET SCORE=0: LET H$="": LET NM=3: L
ET NMT=0: LET LEVEL=0: LET HSCORE=0: LET
 EM=0
  14 LET Q=10: LET P=14: LET LEVEL=LEVEL
+1
  15 LET HM=0: LET VM=0
  16 LET Y(1)=174: LET Y(2)=174: LET Y(3
)=174: LET Y(4)=174
  20 GO SUB 9000
  30 GO SUB 1000
  35 PRINT AT Q,P;"a"
  40 GO SUB 2000
  44 FOR I=1 TO NM
  45 IF Y(I)<24 THEN  GO TO 8000
  46 NEXT I
  47 IF NMT=1 THEN  LET NMT=0: LET NM=NM
+1: GO TO 14
  50 GO SUB 3000
  60 GO TO 40
 999 STOP
1000 REM *ATTACK MISSILE**
1001 FOR K=1 TO NM
1010 LET S(K)=INT (RND*255)+1
1020 LET R(K)=174
1030 LET G(K)=R(K)/(S(K)-TX)
1035 LET T=Y(K)
1040 DEF FN X(T)=(Y(K)+(G(K)*TX))/G(K)
1041 IF QW=1 THEN  LET QW=0: RETURN
1045 NEXT K
1050 RETURN
2000 REM **PLOT MISSILE**
2001 FOR K=1 TO NM
2010 LET Y(K)=Y(K)-4
2020 LET T=Y(K)
```

```
2030 PLOT S(K),174
2031 DRAW INT (FN X(T)-S(K)),Y(K)-174
2035 NEXT K
2040 RETURN
3000 REM **MOVEMENT**
3010 LET A=255-IN 61438
3011 IF A=0 THEN  RETURN
3020 IF A>127 THEN  LET A=A-128
3030 IF A>63 THEN  LET A=A-64
3040 IF A>31 THEN  LET A=A-32
3050 IF A>15 THEN  LET A=A-16: LET HM=-1
3060 IF A>7 THEN  LET A=A-8: LET HM=1
3070 IF A>3 THEN  LET A=A-4: LET VM=1
3090 IF A>1 THEN  LET A=A-2: LET VM=-1
3100 IF A=1 THEN  LET FIRE=1
3110 PRINT AT Q,P;" "
3120 LET P=P+HM: LET Q=Q+VM
3125 INK 0
3126 IF Q<0 THEN  LET Q=0
3127 IF Q>18 THEN  LET Q=18
3128 IF P<0 THEN  LET P=0
3129 IF P>31 THEN  LET P=31
3130 PRINT AT Q,P;"a"
3135 LET HM=0: LET VM=0
3140 IF FIRE=0 THEN  RETURN
3150 LET TAX=P*8+4
3160 LET TAY=(21-Q)*8+4
3165 IF TAX>128 THEN  LET M1=247
3166 IF TAX<=128 THEN  LET M1=9
3170 PLOT M1,16: DRAW TAX-M1,TAY-16
3180 LET FIRE=0
3181 INK 2
3182 PRINT AT Q,P;"b"
3183 FOR Z=1 TO 80: NEXT Z: PRINT AT Q,P
;" "
3200 INK 5
3201 OVER 1
3205 PLOT M1,16
3210 DRAW TAX-M1,TAY-16
```

```
3211 OVER 0
3215 INK 0
3220 FOR K=1 TO NM
3230 IF ABS (TAX-FN X(Y(K)))<4 AND ABS (
TAY-Y(K))<4 THEN  GO TO 3260
3240 NEXT K
3245 PRINT AT Q,P;"a"
3250 RETURN
3305 LET QW=1: GO SUB 1010
3306 LET Y(K)=174
3307 PRINT AT 0,0;S$
3308 LET SCORE=SCORE+1
3309 PRINT #1; AT 0,24;"SCORE=";SCORE
3310 IF SCORE/10=INT (SCORE/10) THEN  LE
T NMT=1
3320 GO TO 3240
8000 REM **DEAD ROUTINE**
8010 CLS
8011 LET LEVEL=0
8015 PRINT "SCORE=";SCORE
8020 FOR I=1 TO 4
8030 FOR J=0 TO 7
8040 PAPER J
8044 BORDER 7-J
8045 CLS
8046 BEEP .009,J*2
8050 NEXT J
8060 NEXT I
8070 BORDER 7
8080 PRINT TAB (10);"GAME OVER"
8090 PRINT TAB (10);"==== ===="
8100 IF SCORE<HSCORE THEN  GO TO 8700
8110 LET HSCORE=SCORE
8115 LET H$=""
8120 LET SCORE=0
8130 PRINT : PRINT
8140 PRINT  FLASH 1;"NEW HIGH SCORE"; FL
ASH 0
8150 FOR J=1 TO 13
```

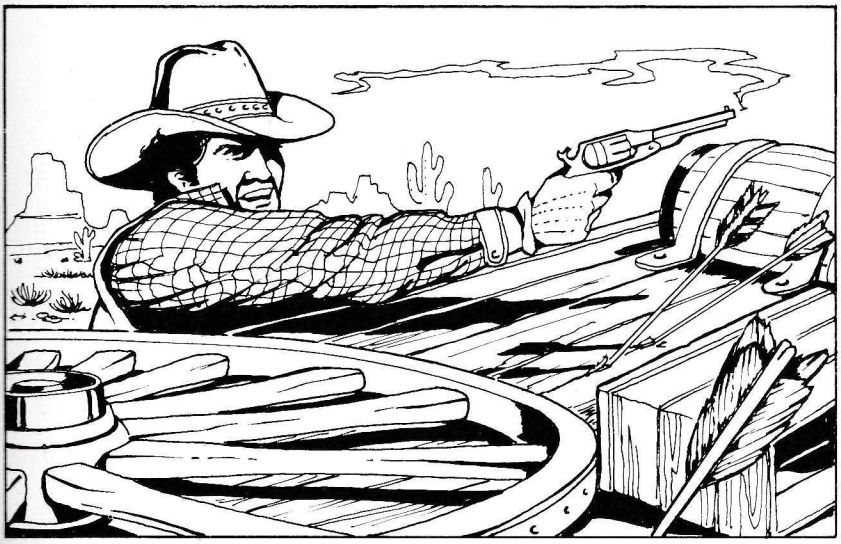
```
8160 PRINT AT 8,(2+(J*2)); INVERSE 1;CHR
$ (64+J); FLASH 0
8170 NEXT J
8180 FOR J=1 TO 13
8190 PRINT AT 12,(2+(J*2)); INVERSE 1;CH
R$ (77+J); INVERSE 0
8200 NEXT J
8210 PRINT AT 16,14; INVERSE 1;"."; INVE
RSE 0
8211 PRINT AT 16,16; INVERSE 1;"#"; INVE
RSE 0
8220 LET J=9: LET I=4
8230 PRINT AT J,I;"^"
8240 GO SUB 8400
8245 IF EM=1 THEN  LET EM=0: GO TO 14
8250 GO TO 8240
8400 REM **MOVE POINTER**
8410 LET L=0: LET R=0: LET F=0
8420 LET A=255-IN 61438
8430 IF A=0 THEN  RETURN
8440 IF A>127 THEN  LET A=A-128
8450 IF A>63 THEN  LET A=A-64
8460 IF A>31 THEN  LET A=A-32
8470 IF A>15 THEN  LET A=A-16: LET L=1
8480 IF A>7 THEN  LET A=A-8: LET R=1
8490 IF A>3 THEN  LET A=A-4
8500 IF A>1 THEN  LET A=A-2
8510 IF A=1 THEN  LET F=1
8520 PRINT AT J,I;" "
8530 IF L=1 THEN  LET I=I-2
8540 IF R=1 THEN  LET I=I+2
8541 IF I=30 AND J=9 THEN  LET I=4: LET
J=13
8542 IF I=30 AND J=13 THEN  LET I=14: LE
T J=17
8543 IF I=18 AND J=17 THEN  LET I=4: LET
 J=9
8544 IF I=2 AND J=9 THEN  LET I=16: LET
J=17
```

```
8545 IF I=2 AND J=13 THEN  LET I=28: LET
 J=9
8546 IF I=12 AND J=17 THEN  LET I=28: LE
T J=13
8550 PRINT AT J,I;"^"
8560 IF F=0 THEN  RETURN
8570 IF J=9 THEN  LET H$=H$+CHR$ (65+((I
/2)-2))
8571 IF J=13 THEN  LET H$=H$+CHR$ (78+((
I/2)-2))
8572 IF J=17 AND I=14 THEN  LET H$=H$+".
"
8573 IF J=17 AND I=16 THEN  LET EM=1
8574 IF LEN (H$)=5 THEN  LET EM=1
8580 PRINT AT 20,12;H$
8590 RETURN
8700 PRINT AT 20,12;"FIRE TO CONTINUE"
8710 IF INKEY$<>"0" THEN  GO TO 8710
8720 LET EM=1
8730 GO TO 14
9000 REM **DRAW SCREEN**
9001 PAPER 5
9002 CLS
9010 FOR I=0 TO 31
9020 PRINT AT 21,I; INK 4;CHR$ 143
9030 NEXT I
9040 INK 0
9050 PRINT AT 20,0;"d": PRINT AT 20,16;".
f"
9060 PRINT AT 20,31;"c"
9070 PRINT AT 20,15;"e"
9080 LET TX=128: LET TY=0
9100 PLOT 143,8
9110 DRAW -32,0,PI
9115 INK 0
9120 PRINT #1;AT 0,0;"LEVEL=";LEVEL
9130 PRINT #1;AT 0,24;"SCORE=";SCORE
9140 PRINT #1;AT 1,19;"HIGH SCORE=";HSCO
RE
```

```
9150 PRINT #1;AT 1,0;"HIGH SCORER=";H$
9160 RETURN
9900 REM **GRAPHICS**
9910 FOR I=0 TO 47
9920 READ A
9930 POKE USR "A"+I,A
9940 NEXT I
9950 DATA 24,24,24,231,231,24,24,24
9960 DATA 60,126,254,254,254,127,127,63
9965 DATA 128,64,60,60,126,126,255,255
9970 DATA 1,2,60,60,126,126,255,255
9975 DATA 0,8,28,92,252,254,255,255
9980 DATA 0,16,56,58,63,127,255,255
9990 RETURN
```

Chapter 5

Shootout



Description

Shootout is a game for two players which requires skill, cunning and ultra-fast reflexes in order to become the greatest gunfighter in the Wild West. The object of the game is to win six duels to the death against your opponent while avoiding the random shots being taken at you by some Indians escaping in their stolen wagon.

This may all sound extremely easy, but just remember that the life expectancy of a gunfighter is very, very short.

Hints on Entry

As the game is totally graphical, plenty of use is made of the user-defined graphics facility and these characters are represented in lower case letters as usual. When the program has been entered it should be saved on tape or microdrive by typing:

```
SAVE "SHOOTOUT" LINE 1
```

or

```
SAVE *"M";1;"SHOOTOUT" LINE 1
```

Techniques

When writing games programs it is often necessary to produce a complex picture on the screen, which as the game progresses changes to suit a developing situation. In such a case, when it is time to start a new game the screen has to be regenerated and this process can often take a long time.

A method of surmounting this problem is to create the picture once at the outset of the program and then to transfer it, using machine code, into a position in high memory. The complete picture is then retained above RAMTOP and can be reproduced on the screen very quickly by using a second routine.

The following short program reorganises the memory, leaving space for the picture, and provides the user with the two required machine code routines.

```
10 CLEAR 54000
20 FOR I=1 TO 23
30 READ A
40 POKE 54000+I,A
50 NEXT I
60 DATA 1,0,27,17,216,214,33,0,64,237,176,201
70 DATA 1,0,27,17,0,64,33,216,204,237,176,201
```

When the program has been entered and run, the system is initialised and the routines are situated at locations 54000 and 54012. Therefore, to store a picture USR 54000 is typed, and to recall a picture USR 54012 is typed.

As the routines are set above RAMTOP, typing NEW will not

affect them and they remain usable until the machine is switched off.

Instructions

Load the program from tape or microdrive by typing:

```
LOAD""
```

or

```
LOAD *"M";1;"SHOOTOUT"
```

When loaded the program will run automatically, and a brief description of the game will appear on the screen. Each duel will last for a set period of time, the passing of which is indicated by the descending scale at each end of the screen. Each player must attempt to move his character into the correct position in order to kill his opponent. The score obtained for a kill depends on the amount of time remaining.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE REPRESENTED BY LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
  1 POKE 23658,8
  5 GO SUB 5000
  6 LET Q$=CHR$ 138: FOR I=1 TO 31: LET
 Q$=Q$+CHR$ 138: NEXT I
 10 INK 0: PAPER 6: CLS : BORDER 6: INP
UT ""
 15 LET D1=3: LET D2=3
 16 LET S1=0: LET S2=0
 17 PRINT #0;TAB 8; FLASH 1;"PRESS ANY
KEY"
```

```
  20 PRINT '''''''''' INK 2;                "
  ";CHR$ 139;CHR$ 131;" ";CHR$ 138;CHR$
133;" ";CHR$ 139;CHR$ 135;" ";CHR$ 138;"
 ";CHR$ 138;"  ";CHR$ 139;CHR$ 134;" ";C
HR$ 139;CHR$ 135;" ";CHR$ 138;" ";CHR$ 1
38;CHR$ 142;" ";CHR$ 138;"        ";CHR$ 1
31;CHR$ 135;" ";CHR$ 139;CHR$ 135;" ";CH
R$ 138;CHR$ 133;" ";CHR$ 138;CHR$ 136;CH
R$ 138;"  ";CHR$ 138;CHR$ 133;" ";CHR$ 1
38;CHR$ 133;" ";CHR$ 138;CHR$ 136;CHR$ 1
38;CHR$ 138;CHR$ 138;CHR$ 138;"       ";C
HR$ 140;CHR$ 141;" ";CHR$ 138;CHR$ 133;"
 ";CHR$ 142;CHR$ 141;" ";CHR$ 134;CHR$ 1
34;CHR$ 130;"  ";CHR$ 142;CHR$ 137;" ";C
HR$ 142;CHR$ 141;" ";CHR$ 134;CHR$ 134;C
HR$ 130;CHR$ 138;CHR$ 129;CHR$ 138
  25 PRINT  FLASH 1; INK 2; PAPER 1;AT 0
,0;Q$;AT 21,0;Q$: FOR N=1 TO 20: PRINT
FLASH 1; INK 2; PAPER 1;AT N,0;"?";AT N,
31;"?": NEXT N
  30 LET L=1: LET C=3: GO SUB 140
  35 LET L=16: LET C=28: GO SUB 140
  40 PRINT AT 3,25;"b";AT 18,6;"a";AT 2,
17;"kl";AT 3,17;"mn";AT 15,12;"kl";AT 16
,12;"mn";AT 18,19;"p";AT 19,19;"jo";AT 4
,8;"p";AT 5,8;"jo"
  50 RESTORE 60: FOR M=1 TO 4: FOR N=1 T
O 9: READ F: BEEP .5,F: IF INKEY$="" THE
N  NEXT N: READ F: BEEP 1,F: PAUSE 10: I
F INKEY$="" THEN  NEXT M
  60 DATA -12,-9,-7,-5,0,-2,-9,-4,-4,-5,
-5,-5,-3,-2,-2,2,0,-2,-3,-2,3,3,2,0,-5,0
,-2,-4,-5,-4,-4,-5,-7,-5,-12,-7,-9,-10,-
10,-12
  95 CLS
 100 LET L1=18: LET L2=3
 110 LET W=0: LET WW=1
 120 PRINT AT L1,3;"a";AT L2,28;"b"
 130 GO TO 150
```

```
 140 PRINT  INK 4;AT L,C;"f";AT L+1,C;"c
h";AT L+2,C-1;"hei";AT L+3,C-1;"gd";AT L
+4,C;"c": RETURN
 150 LET L=16: LET C=9: GO SUB 140
 160 LET L=1: LET C=22: GO SUB 140
 170 LET T=175
 180 INK 2: FOR Y=0 TO 175 STEP 2: PLOT
0,Y: DRAW 2,0: PLOT 253,Y: DRAW 2,0: BEE
P .003,Y/3: NEXT Y: INK 0
 200 LET K1=(IN 64510<>255)+2*(IN 65022<
>255)+4*(IN 65278<>255)
 205 INPUT "": LET BONUS=INT (200*RND):
PRINT #0;TAB 12;"BONUS:";BONUS
 210 LET K2=(IN 57342<>255)+2*(IN 49150<
>255)+4*(IN 32766<>255)
 215 LET D=0
 220 IF K1>=4 THEN  GO SUB 1000
 225 IF D THEN  GO TO 3000
 230 IF K2>=4 THEN  GO SUB 2000
 235 IF D THEN  GO TO 3000
 240 IF NOT K1 OR K1=3 THEN  GO TO 270
 250 IF K1=1 AND L1>0 THEN  PRINT AT L1,
3;" ": LET L1=L1-1: PRINT AT L1,3;"a"
 260 IF K1=2 AND L1<21 THEN  PRINT AT L1
,3;" ": LET L1=L1+1: PRINT AT L1,3;"a"
 270 IF NOT K2 OR K2=3 THEN  GO TO 300
 280 IF K2=1 AND L2>0 THEN  PRINT AT L2,
28;" ": LET L2=L2-1: PRINT AT L2,28;"b"
 290 IF K2=2 AND L2<21 THEN  PRINT AT L2
,28;" ": LET L2=L2+1: PRINT AT L2,28;"b"
 300 LET W=W+WW
 310 IF W<0 THEN  LET W=0: LET WW=1
 320 IF W>18 THEN  LET W=18: LET WW=-1
 330 PRINT AT W,13;"  ";AT W+1,13;"kl";A
T W+2,13;"mn";AT W+3,13;"  ";AT 18-W,17;
"  ";AT 19-W,17;"kl";AT 20-W,17;"mn";AT
21-W,17;"  "
 331 IF W<15 AND RND>.9 THEN  LET Y=(20-
W)*8: PLOT 104,Y: DRAW -72,0: INVERSE 1:
```

```
 PLOT 104,Y: DRAW -72,0: INVERSE 0: IF W
+1=L1 THEN  LET D=1: GO TO 3000
 332 IF W<14 AND RND>.9 THEN  LET Y=(W+2
)*8: PLOT 152,Y: DRAW 72,0: INVERSE 1: P
LOT 152,Y: DRAW 72,0: INVERSE 0: IF 19-W
=L2 THEN  LET D=2: GO TO 3000
 340 LET T=T-1
 350 IF T<0 THEN  GO TO 6000
 360 INVERSE 1: INK 2
 370 PLOT 0,T: DRAW 2,0: PLOT 253,T: DRA
W 2,0
 380 INVERSE 0: INK 0
 390 GO TO 200
1000 LET K1=K1-4
1005 BEEP .005,0
1010 LET X=200
1020 IF L1=1 OR L1=2 OR L1=5 THEN  LET X
=144
1030 IF L1=3 OR L1=4 THEN  LET X=136
1040 IF L1=19-W OR L1=20-W THEN  LET X=1
04
1050 IF L1=W+1 OR L1=W+2 THEN  LET X=72
1060 IF L1=16 OR L1=17 OR L1=20 THEN  LE
T X=40
1070 IF L1=18 OR L1=19 THEN  LET X=32
1100 LET Y=(21-L1)*8+4
1110 PLOT 31,Y: DRAW X,0
1120 INVERSE 1
1130 PLOT 31,Y: DRAW X,0
1140 INVERSE 0
1150 IF L1=L2 AND X=200 THEN  LET D=2
1160 RETURN
2000 LET K2=K2-4
2005 BEEP .005,0
2010 LET X=200
2020 IF L2=16 OR L2=19 OR L2=20 THEN  LE
T X=144
2030 IF L2=17 OR L2=18 THEN  LET X=136
2040 IF L2=W+1 OR L2=W+2 THEN  LET X=104
```

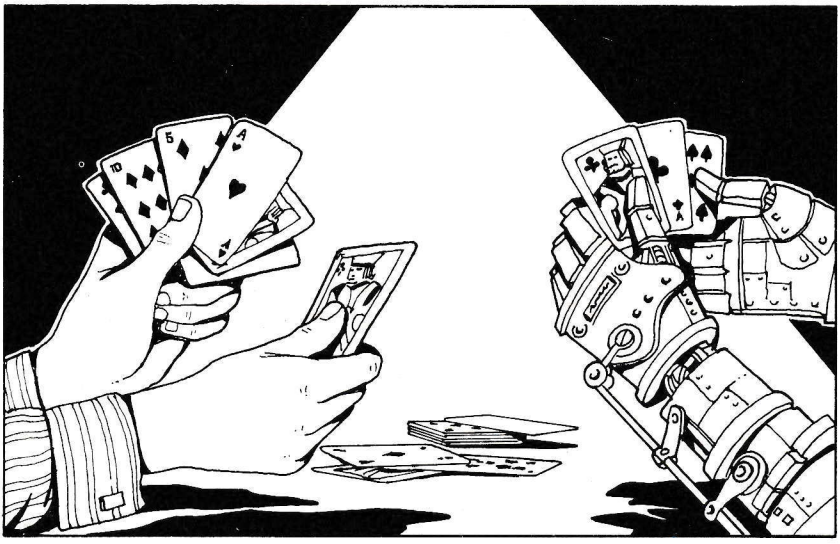
```
2050 IF L2=19-W OR L2=20-W THEN  LET X=7
2
2060 IF L2=1 OR L2=4 OR L2=5 THEN  LET X
=40
2070 IF L2=2 OR L2=3 THEN  LET X=32
2100 LET Y=(21-L2)*8+4
2110 PLOT 224,Y: DRAW -X,0
2120 INVERSE 1
2130 PLOT 224,Y: DRAW -X,0
2140 INVERSE 0
2150 IF L2=L1 AND X=200 THEN  LET D=1
2160 RETURN
3000 IF L1=21 THEN  LET L1=20
3005 IF D=1 THEN  LET S2=S2+T+BONUS
3010 IF L2=21 THEN  LET L2=20
3015 IF D=2 THEN  LET S1=S1+T+BONUS
3020 IF D=1 THEN  PRINT AT L1,3;"p";AT L
1+1,3;"jo": IF L1-1>=0 THEN  FOR N=L1-1
TO 0 STEP -1: BEEP .05,21-N: PRINT AT N,
3;"q": FOR I=1 TO 25: NEXT I: PRINT AT N
,3;" ": NEXT N
3030 IF D=2 THEN  PRINT AT L2,28;"p";AT
L2+1,28;"jo": IF L2-1>=0 THEN  FOR N=L2-
1 TO 0 STEP -1: BEEP .05,21-N: PRINT AT
N,28;"q": FOR I=1 TO 25: NEXT I: PRINT A
T N,28;" ": NEXT N
3040 RESTORE 3040: FOR N=1 TO 11: READ E
,F: BEEP E/2,F: NEXT N: DATA 1,-12,.75,-
12,.25,-12,1,-12,.75,-9,.25,-10,.75,-10,
.25,-12,.75,-12,.25,-13,2,-12
3050 IF D=1 THEN  LET D1=D1-1: IF D1=0 T
HEN  GO TO 4000
3060 IF D=2 THEN  LET D2=D2-1: IF D2=0 T
HEN  GO TO 4000
3065 CLS
3070 PRINT AT 10,7;"PLAYER 1";AT 10,20;"
PLAYER 2";AT 12,0;"LIVES";AT 14,0;"SCORE
";AT 12,10;D1;AT 12,25;D2;AT 14,10-(S1>9
9);S1;AT 14,25-(S2>99);S2
```

```
3080 FOR I=60 TO 0 STEP -.5: BEEP .003,I
: NEXT I
3090 GO TO 95
4000 PRINT AT 10,12; INVERSE 1;"GAME OVE
R!"
4010 FOR I=1 TO 200: NEXT I
4020 CLS
4030 PRINT "WINNER :";
4040 IF D=2 THEN  PRINT "PLAYER 1": PRIN
T : PRINT "SCORE :";S1
4050 IF D=1 THEN  PRINT "PLAYER 2": PRIN
T : PRINT "SCORE :";S2
4060 FOR I=0 TO 60 STEP .5: BEEP .003,I:
 NEXT I
4070 GO TO 10
5000 RESTORE 5100
5010 FOR N=USR "A" TO USR "Q"+7
5020 READ Z: POKE N,Z
5030 NEXT N
5040 RETURN
5100 DATA BIN 00001000,BIN 00011100,BIN
00001000,BIN 00001111,BIN 00001000,BIN 0
0001100,BIN 00010100,BIN 00110100
5110 DATA BIN 00010000,BIN 00111000,BIN
00010000,BIN 11110000,BIN 00010000,BIN 0
0110000,BIN 00101000,BIN 00101100
5120 DATA 126,126,126,126,126,126,126,12
6
5130 DATA 126,126,254,254,254,254,126,12
6
5140 DATA 126,126,127,127,127,127,126,12
6
5150 DATA 0,0,BIN 00011000,BIN 00111100,
126,126,126,126
5160 DATA 60,62,63,63,63,BIN 00011111,0,
0
5170 DATA 0,0,BIN 00011000,60,60,60,60,6
0
```

```
5180 DATA 60,BIN 01111100,252,252,252,BI
N 11111000,0,0
5190 DATA 0,BIN 11101011,BIN 10101010,BI
N 10101010,BIN 11101011,BIN 11001010,BIN
 10101010,BIN 10101010
5200 DATA BIN 00000001,BIN 00000110,BIN
00001000,BIN 00010000,BIN 00010000,BIN 0
0100000,BIN 00100000,BIN 00100000
5210 DATA BIN 10000000,BIN 01100000,BIN
00010000,BIN 00001000,BIN 00001000,BIN 0
0000100,BIN 00000100,BIN 00000100
5220 DATA BIN 00011100,BIN 00011111,BIN
00101111,BIN 00101111,BIN 00111111,BIN 0
0100000,BIN 00100000,0
5230 DATA BIN 00111000,BIN 11111000,BIN
11110100,BIN 11110100,BIN 11111100,BIN 0
0000100,BIN 00000100,0
5240 DATA 0,128,128,128,128,0,0,0
5250 DATA 8,8,BIN 00111110,8,8,8,BIN 000
11100,BIN 00111110
5260 DATA 0,0,0,BIN 01011001,255,BIN 010
00000,0,0
6000 PRINT AT 10,13; INVERSE 1;"SUNSET!"
6010 FOR I=1 TO 200: NEXT I
6020 CLS
6030 LET d1=d1-1: IF d1=0 THEN  LET d=1:
 GO TO 4000
6040 LET d2=d2-1: IF d2=0 THEN  LET d=2:
 GO TO 4000
6050 GO TO 3070
```

Chapter 6

Cribbage



Description

If you are a card-playing fanatic then you will love the opportunity to test your ability against a computer in this, the most skilful of games for two players.

As the game progresses, it should become obvious that the computer is no easy adversary as the program is designed to detect many of the crafty tricks you may play for extra points and the computer also knows a few tricks of its own. So good luck, play fair and with a lot of practice you might just win a game—if you're lucky!

Hints on Entry

Although the program is very long, there are no major areas which are likely to cause problems. The user-defined graphics, which are few and far between, are represented as usual by lower case letters. The major part of the program consists of the mathematical algorithms used to enable the computer to play an intelligent game, and a series of validation routines to make sure that you, the computer's opponent, are abiding by the rules of the game.

When the program has been entered, it should be saved on tape or microdrive in the normal way using:

```
SAVE "CRIB" LINE 1
```

or

```
SAVE *"M";1;"CRIB" LINE 1
```

The program has been written using a modular approach and consists of a large number of subroutines each with an associated REM statement. If after entering and running the program a particular facility does not operate, then it should be a simple task to locate and amend the error.

Techniques

The word technique is often used to describe the way in which a program has been constructed. A program which has been written using a good technique will be clear, easy to follow and will generally run faster than one which has been designed in a haphazard manner.

One of the best techniques to employ in program writing is that known as modular programming, in which the program is built up using a series of blocks or modules.

1) The program is divided into a series of modules by constructing an algorithm or flow diagram such as that shown in Fig. 2.
2) When a rough outline has been completed, each of the blocks can be coded into a subroutine to be used in the program. Each routine should contain at least one REM

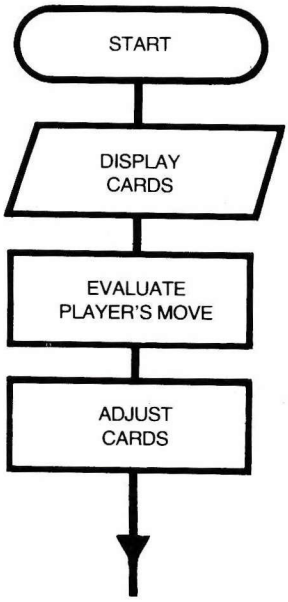


Fig. 2

statement indicating the purpose of that section for future reference.

As mentioned above, a program written in a logical and structured manner will operate more efficiently and will be far more rewarding to the programmer.

Instructions

When the game is loaded from tape or microdrive, it will run automatically. The cards will be cut to see who has first crib, and the computer will deal out the cards.

No playing instructions are included at this point as it is assumed that the rules of the game and the *modus operandi* are understood. To make operation as easy and as fool-proof as possible, the user refers to his cards by number and if at any time a card cannot be played then the player types P to pass and the computer takes its turn.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED BY LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED .

```
   1  LET RC=0: LET RP=0: LET GO=INT (RN
D*2)+1: INK 0: LET FL=0
   2 LET N$="
                              ": LET W$="
    "
   3 FOR I=1 TO 3
   4 LET N$=N$+N$
   5 NEXT I
   6 LET SC1=5: LET SC2=170: LET SP1=5:
LET SP2=125
   7 LET SCOREP=0: LET SCOREC=0
   8 RANDOMIZE
   9 POKE 23658,8
  10 CLS
  15 GO SUB 9900
  20 DIM C(5): DIM G$(8,3): DIM D$(4,3):
 DIM E$(4,3): DIM P(8): DIM I$(5,3): DIM
 F$(1,3): DIM A$(12,3): DIM H$(5,3): DIM
 B$(12,3): DIM P$(5,3): DIM Q$(8,1): DIM
 I(10): DIM R$(8,1): DIM P$(5,3): DIM C$
(5,3)
  21 LET F$(1)="999"
  29 GO SUB 4000
  30 PRINT AT 1,23;"DEALING"
  31 PRINT AT 3,24;"CARDS"
  32 GO SUB 9000
  33 IF GO/2<>INT (GO/2) THEN  PRINT #1;
"PLAYER'S CRIB"
```

```
  34 IF GO/2=INT (GO/2) THEN  PRINT #1;"
COMPUTER'S CRIB"
  35 LET S=1: LET F=6: GO SUB 9130
  36 LET S=7: LET F=12: GO SUB 9130: GO
SUB 9400
  37 FOR I=1 TO 6: LET A$(I)=B$(I+6): NE
XT I: LET CN=6: LET C1=1: LET C2=6: GO S
UB 6000
  38 GO SUB 6100
  39 GO SUB 6200
  41 GO SUB 5000
  42 LET X1=192: LET Y1=104: GO SUB 6300
  43 GO SUB 6400
  44 PRINT AT 1,21;" COMPUTER";AT 2,21;"
 THINKING";AT 3,21;"PLEASE WAIT": GO SUB
 9560: FOR I=1 TO 6: LET A$(I)=B$(I): NE
XT I: GO SUB 8000
  45 PRINT AT 1,21;"ENTER CARD"
  46 PRINT AT 2,20;"TO BE PLACED"
  47 PRINT AT 3,21;"  IN BOX    "
  48 GO SUB 9560
  49 LET C=6: GO SUB 5200
  50 LET M=3
  51 LET Z$=INKEY$: IF Z$="" THEN  GO TO
 51
  52 LET Z=VAL Z$: IF Z>6 OR Z<1 THEN  G
O TO 51
  53 LET C$(M)=A$(Z+6)
  54 IF C$(M)=C$(3) AND M=4 THEN  GO TO
51
  55 PRINT AT 4,22+((M-3)*2); INK I(Z);R
$(Z);AT 5,22+((M-3)*2);Q$(Z): LET M=M+1
  56 IF M<>5 THEN  GO TO 51
  57 LET K=1
  58 FOR I=7 TO 12
  60 IF A$(I)<>C$(3) AND A$(I)<>C$(4) TH
EN  LET P$(K)=A$(I): LET K=K+1
  62 NEXT I
  63 GO SUB 5300
```

```
  70 FOR I=1 TO 4: LET A$(I)=P$(I): NEXT
 I
  80 LET C1=1: LET C2=4: LET CN=4: GO SU
B 6000: GO SUB 6100: GO SUB 6200:: GO SU
B 5000
  90 INK 0: PRINT AT 1,20;" PRESS ENTER"
;AT 2,20;"   TO CUT   ";AT 3,20;"  THE C
ARDS";AT 4,20;"        ";AT 5,20;"
 ": GO SUB 9560
  91 IF INKEY$="" THEN  GO TO 91
 100 LET A=INT (RND*4)+1
 110 LET F$(1)=S$(A)
 120 LET A=INT (RND*26)+1
 130 IF A/2=INT (A/2) THEN  GO TO 120
 140 LET F$(1,2 TO 3)=V$(A TO A+1)
 150 FOR I=1 TO 12
 160 IF B$(I)=F$(1) THEN  GO TO 100
 170 NEXT I
 180 GO SUB 5400
 185 LET X1=192: LET Y1=104: GO SUB 6300
 189 LET C1=1: LET C2=1: LET CN=1: LET A
$(1)=F$(1): GO SUB 6100: LET Y$=Q$(1): L
ET Z$=R$(1): LET II=25: GO SUB 5005
 190 IF F$(1,2 TO 3)="11" AND GO/2=INT (
GO/2) THEN  LET SOCREP=SCOREP+2: GO SUB
4000
 191 IF F$(1,2 TO 3)="11" AND GO/2<>INT
(GO/2) THEN  LET SOCREC=SCOREC+2: GO SUB
 4000
 192 GO SUB 5600: GO SUB 5500: GO SUB 41
00: GO SUB 5600: IF GO/2<>INT (GO/2) THE
N  LET FL=1: GO TO 290
 193 GO SUB 5600: GO SUB 5500: LET C1=1:
 LET C2=1: LET CN=1: LET A$(1)=F$(1): GO
 SUB 6100: LET Y$=Q$(1): LET Z$=R$(1): L
ET II=25: GO SUB 5005: LET X1=192: LET Y
1=104: GO SUB 6300
 194 FOR I=1 TO 4: LET A$(I)=P$(I): NEXT
 I
```

```
 195 LET C1=1: LET C2=4: LET CN=4
 196 GO SUB 6000: GO SUB 6100: GO SUB 62
00: GO SUB 5000
 197 LET X1=192: LET Y1=104: GO SUB 6300
 200 FOR I=1 TO 4
 201 IF VAL (F$(1,2 TO 3))>VAL (P$(I,2 T
O 3)) THEN  NEXT I
 204 FOR J=5 TO I+1 STEP -1
 205 LET P$(J)=P$(J-1)
 207 NEXT J
 208 LET P$(I)=F$(1)
 210 LET I$(5)=F$(1)
 220 LET C$(5)=F$(1)
 230 GO SUB 5500
 235 INK 0
 240 PRINT AT 1,20;" PLAYERS";AT 2,20;"
SCORE="
 245 GO SUB 9560
 250 FOR I=1 TO 5: LET H$(I)=P$(I): NEXT
 I: GO SUB 7000
 255 INK 0
 260 PRINT AT 2,27;S
 270 LET SCOREP=SCOREP+S
 275 LET SP1=SCOREP*5
 280 GO SUB 4000
 285 IF FL=1 THEN  LET FL=0: GO TO 376
 290 GO SUB 5500
 300 PRINT AT 1,20;" COMPUTERS";AT 2,20;
" SCORE="
 301 GO SUB 5600: LET C1=1: LET C2=1: LE
T CN=1: LET A$(1)=F$(1): GO SUB 6100: LE
T Y$=Q$(1): LET Z$=R$(1): LET II=25: GO
SUB 5005: LET X1=192: LET Y1=104: GO SUB
 6300
 310 GO SUB 9560
 311 GO SUB 5300
 312 FOR I=1 TO 4
 313 LET A$(I)=I$(I)
 314 NEXT I
```

```
 315 LET C1=1: LET C2=4: LET CN=4
 316 GO SUB 6000: GO SUB 6100: GO SUB 62
00: GO SUB 5000
 320 FOR I=1 TO 5: LET H$(I)=I$(I): NEXT
 I
 321 FOR I=1 TO 4
 322 IF VAL F$(1,2 TO 3)>VAL (H$(I,2 TO
3)) THEN  NEXT I
 323 FOR J=5 TO I+1 STEP -1
 324 LET H$(J)=H$(J-1)
 325 NEXT J
 326 LET H$(I)=F$(1)
 327 GO SUB 7000
 328 INK 0
 330 PRINT AT 2,27;S
 340 LET SCOREC=SCOREC+S
 350 LET SC1=SCOREC*5
 360 GO SUB 4000
 370 GO SUB 5500
 375 IF FL=1 THEN  GO TO 193
 376 GO SUB 5500: IF GO/2<>INT (GO/2) TH
EN  PRINT AT 1,21;"PLAYER'S"
 377 IF GO/2=INT (GO/2) THEN  PRINT AT 1
,21;"COMPUTER'S"
 380 PRINT AT 2,21;"CRIBBAGE";AT 3,21;"S
CORE=";
 390 GO SUB 9560
 400 GO SUB 5300
 410 FOR I=1 TO 4
 420 LET A$(I)=C$(I)
 430 NEXT I
 440 LET C1=1: LET C2=4: LET CN=4
 450 GO SUB 6000: GO SUB 6100: GO SUB 62
00: GO SUB 5000
 460 FOR I=1 TO 5
 470 LET B$(I)=C$(I)
 480 NEXT I
 490 LET S=1: LET F=5: GO SUB 9155
 500 FOR I=1 TO 5
```

```
 510 LET H$(I)=A$(I)
 520 NEXT I
 530 GO SUB 7000
 540 PRINT AT 3,27;S
 550 IF GO/2=INT (GO/2) THEN  LET SCOREC
=SCOREC+S: LET SC1=SCOREC*5: GO TO 570
 560 LET SCOREP=SCOREP+S: LET SP1=SCOREP
*5
 570 GO SUB 4000
 580 LET GO=GO+1
 790 CLS
 800 GO TO 29
 990 STOP
4000 REM **SCORE BOARD**
4001 LET SP1=SCOREP*5
4002 LET SC1=SCOREC*5
4003 INK 0
4010 FOR J=0 TO 7
4020 PRINT AT J,0;"                    "
4030 NEXT J
4032 IF SCOREC>=61 AND RC=1 THEN  LET Z$
="COMPUTER ": GO TO 8300
4034 IF SCOREP>=61 AND RP=1 THEN  LET Z$
="PLAYER": GO TO 8300
4035 GO SUB 9500
4036 INK 0
4037 IF SCOREP>60 THEN  LET SCOREP=SCORE
P-60: LET RP=1: LET SP1=SCOREP*5: LET SP
2=125
4038 IF SCOREC>60 THEN  LET SCOREC=SCORE
C-60: LET RC=1: LET SC1=SCOREC*5: LET SC
2=170
4041 IF SCOREC>30 THEN  LET SC2=159: LET
 SC1=(150-((SCOREC-31)*5))
4042 IF SCOREP>30 THEN  LET SP2=136: LET
 SP1=(150-((SCOREP-31)*5))
4050 PLOT SC1,SC2: PLOT SC1-1,SC2+1: PLO
T SC1,SC2+1: PLOT SC1+1,SC2+1: PLOT SC1,
SC2+2: PLOT SC1,SC2+3
```

```
4060 PLOT SP1,SP2: PLOT SP1-1,SP2-1: PLO
T SP1,SP2-1: PLOT SP1+1,SP2-1: PLOT SP1,
SP2-2: PLOT SP1,SP2-3
4070 RETURN
4100 REM *PLAY ROUTINE*
4101 LET PO=GO+1: LET PASS=0: LET CP=0:
LET TCP=0: LET TOTAL=0
4102 INK 0
4110 FOR I=1 TO 4
4120 LET D$(I)=P$(I)
4130 LET E$(I)=I$(I)
4140 NEXT I
4150 PRINT AT 1,20;" PLAY FOR "
4160 PRINT AT 2,20;" POINTS."
4161 PRINT AT 3,20;" PRESS ENTER"
4162 PRINT AT 4,20;" TO CONTINUE"
4163 GO SUB 9560
4164 LET Z$=INKEY$
4165 IF Z$="" THEN  GO TO 4164
4180 IF PO/2<>(INT (PO/2)) THEN  GO SUB
4500
4181 IF TOTAL=31 THEN  LET TOTAL=0: LET
PASS=0: LET CP=0: GO SUB 5600: GO TO 418
8
4182 IF PASS=2 THEN  LET SCOREP=SCOREP+1
: GO SUB 4000: GO SUB 5600: LET TOTAL=0:
 LET CP=0: LET PASS=0: GO TO 4188
4183 IF TCP=8 THEN  GO TO 4190
4184 IF PO/2=INT (PO/2) THEN  GO SUB 460
0
4186 IF TOTAL=31 THEN  LET TOTAL=0: LET
PASS=0: LET CP=0: GO SUB 5600: GO TO 418
8
4187 IF PASS=2 THEN  LET SCOREC=SCOREC+1
: GO SUB 4000: GO SUB 5600: LET TOTAL=0:
 LET CP=0: LET PASS=0
4188 IF TCP<8 THEN  GO TO 4180
4190 IF PO/2=INT (PO/2) THEN  LET SCOREP
=SCOREP+1: GO SUB 4000
4192 IF PO/2<>INT (PO/2) THEN  LET SCORE
```

```
C=SCOREC+1: GO SUB 4000
4200 RETURN
4500 REM *PLAYERS GO*
4501 GO SUB 5500: INK 0
4502 PRINT AT 1,20;" YOUR PLAY:"
4503 GO SUB 9560
4504 FOR I=1 TO 4: LET A$(I)=D$(I): NEXT
 I: LET C1=1: LET C2=4: GO SUB 6100
4520 FOR I=1 TO 4
4525 IF D$(I)="999" THEN  GO TO 4550
4526 IF Q$(I)="f" OR Q$(I)="d" THEN  INK
 2
4530 PRINT AT 3,(20+(2*I));Q$(I,1)
4540 PRINT AT 4,(20+(2*I));R$(I)
4541 INK 0
4545 PRINT AT 5,(20+(2*I)); FLASH 1;I; F
LASH 0
4550 NEXT I
4551 PAUSE 0
4552 LET Y$=INKEY$: IF Y$="P" THEN  GO T
O 4554
4553 IF Y$="" OR CODE (Y$)>57 THEN  GO T
O 4552
4554 IF Y$="P" THEN  LET PO=PO+1: LET PA
SS=PASS+1: RETURN
4555 LET Y=VAL (Y$): IF Y>4 THEN  GO TO
4552
4556 IF D$(Y)="999" THEN  GO TO 4552
4558 LET VL=VAL A$(Y,2 TO 3): IF VL>10 T
HEN  LET VL=10
4559 IF TOTAL+VL>31 THEN  GO TO 4551
4561 LET TOTAL=TOTAL+VL
4562 LET P(CP+1)=VL
4563 LET PO=PO+1
4565 LET CP=CP+1
4570 LET TCP=TCP+1
4571 LET G$(CP)=D$(Y)
4580 LET D$(Y)="999"
4581 FOR I=1 TO CP
4582 LET A$(I)=G$(I)
```

```
4583 NEXT I
4584 GO SUB 5600
4585 LET C1=1: LET C2=CP: LET CN=CP: GO
SUB 6000
4586 GO SUB 6100
4587 GO SUB 6200
4588 GO SUB 5000
4590 GO SUB 4900
4599 RETURN
4600 REM *COMPUTERS GO*
4605 FOR I=1 TO 4: LET A$(I)=E$(I): NEXT
 I: LET C1=1: LET C2=4: GO SUB 6100
4610 GO SUB 5500: INK 0
4620 PRINT AT 1,20;" COMPUTER'S "
4630 PRINT AT 2,20;" PLAY:"
4640 PRINT AT 3,20;" PLEASE WAIT"
4650 GO SUB 9560
4660 FOR Y=1 TO 4
4670 LET VL=VAL (E$(Y,2 TO 3))
4671 IF VL=99 THEN  GO TO 4690
4672 IF VL>10 THEN  LET VL=10
4680 IF TOTAL+VL=15 OR TOTAL+VL=31 THEN
 GO TO 4750
4681 IF CP>=1 THEN  IF VL=P(CP) AND TOTA
L+VL<=31 THEN  GO TO 4750
4690 NEXT Y
4700 FOR Y=4 TO 1 STEP -1
4710 IF E$(Y)="999" THEN  GO TO 4720
4711 LET VL=VAL (E$(Y,2 TO 3))
4712 IF VL>10 THEN  LET VL=10
4715 IF TOTAL+VL<=31 THEN  GO TO 4750
4720 NEXT Y
4749 LET PO=PO+1: LET PASS=PASS+1: RETUR
N
4750 REM *PLAY CARD*
4755 LET P(CP+1)=VL
4760 LET CP=CP+1
4770 LET G$(CP)=E$(Y)
4780 LET E$(Y)="999"
4790 LET TCP=TCP+1: LET TOTAL=TOTAL+VL
4791 LET PO=PO+1
4795 FOR I=1 TO CP
```

```
4796 LET A$(I)=G$(I)
4797 NEXT I
4798 GO SUB 5600
4799 LET C1=1: LET C2=CP: LET CN=CP
4800 GO SUB 6000
4810 GO SUB 6100
4820 GO SUB 6200
4830 GO SUB 5000
4835 GO SUB 4900
4840 RETURN
4900 REM *SCORE EVALUATION*
4901 LET SPG=0: LET TT=0: LET PASS=0
4902 IF TOTAL=15 OR TOTAL=31 THEN  LET S
PG=2
4904 IF CP<=2 THEN  GO TO 4942
4906 FOR I=1 TO CP
4908 LET P(I)=VAL (G$(I,2 TO 3))
4910 NEXT I
4920 LET MAX=P(CP): LET MIN=P(CP)
4922 FOR I=1 TO CP-2
4923 LET TT=0: LET MAX=P(CP): LET MIN=P(
CP)
4924 FOR J=CP TO I STEP -1
4926 IF P(J)>MAX THEN  LET MAX=P(J)
4928 IF P(J)<MIN THEN  LET MIN=P(J)
4930 LET TT=TT+P(J)
4932 NEXT J
4934 LET AV1=(MAX+MIN)/2
4936 LET AV2=TT/((CP-I)+1)
4938 IF AV1=AV2 AND MAX-MIN=CP-I THEN  L
ET SPG=SPG+(CP+1)-I: GO TO 4942
4940 NEXT I
4942 IF CP>=3 THEN  IF P(CP)=P(CP-1) AND
 P(CP)=P(CP-2) THEN  LET SPG=SPG+6: GO T
O 4980
4946 IF CP>=2 THEN  IF P(CP)=P(CP-1) THE
N  LET SPG=SPG+2
4980 IF PO/2=INT (PO/2) THEN  LET SCOREP
=SCOREP+SPG
4982 IF PO/2<>INT (PO/2) THEN  LET SCORE
C=SCOREC+SPG
```

```
4984 GO SUB 4000
4986 RETURN
5000 REM **DRAW SINGLE CARD**
5001 LET JJ=10: LET II=(3*CN)-2: LET Y$=
Q$(C2): LET Z$=R$(C2)
5005 IF Y$="c" OR Y$="d" THEN  INK 0
5006 IF Y$="f" OR Y$="g" THEN  INK 2
5010 PRINT AT JJ,II;Z$
5020 PRINT AT JJ-1,II;Y$
5030 PRINT AT JJ+8,II+4;Z$
5040 PRINT AT JJ+9,II+4;Y$
5050 IF Z$="J" THEN  PRINT AT JJ+3,II+3;
CHR$ 133;AT JJ+4,II+1;CHR$ 136;CHR$ 133;
AT JJ+5,II+1;CHR$ 134;CHR$ 140;CHR$ 137
5060 IF Z$="Q" THEN  PRINT AT JJ+3,II+1;
CHR$ 137;CHR$ 131;CHR$ 134;AT JJ+4,II+1;
CHR$ 138;CHR$ 132;CHR$ 133;AT JJ+5,II+1;
CHR$ 134;CHR$ 140;CHR$ 139;AT JJ+6,II+3;
CHR$ 129
5070 IF Z$="K" THEN  PRINT AT JJ+3,II+1;
CHR$ 138;CHR$ 132;CHR$ 130;AT JJ+4,II+1;
CHR$ 139;CHR$ 134;AT JJ+5,II+1;CHR$ 138;
" ";CHR$ 134
5080 IF Z$="h" OR Z$="9" OR Z$="8" THEN
 PRINT AT JJ+1,II+1;Y$;" ";Y$;AT JJ+3,II
+1;Y$;" ";Y$;AT JJ+5,II+1;Y$;" ";Y$;AT J
J+7,II+1;Y$;" ";Y$
5081 IF Z$="h" THEN  PRINT AT JJ+2,II+2;
Y$;AT JJ+6,II+2;Y$
5082 IF Z$="9" THEN  PRINT AT JJ+4,II+2;
Y$
5090 IF Z$="6" OR Z$="7" THEN  PRINT AT
JJ+1,II+1;Y$;" ";Y$;AT JJ+4,II+1;Y$;" ";
Y$;AT JJ+7,II+1;Y$;" ";Y$
5091 IF Z$="7" THEN  PRINT AT JJ+4,II+2;
Y$
5100 IF Z$="5" OR Z$="4" THEN  PRINT AT
JJ+2,II+1;Y$;" ";Y$;AT JJ+6,II+1;Y$;" ";
Y$
5101 IF Z$="5" THEN  PRINT AT JJ+4,II+2;
Y$
```

```
5110 IF Z$="2" OR Z$="3" THEN  PRINT AT
JJ+2,II+2;Y$;AT JJ+6,II+2;Y$
5111 IF Z$="3" OR Z$="1" THEN  PRINT AT
JJ+4,II+2;Y$
5120 RETURN
5200 REM **NUMBER CARDS**
5210 FOR I=1 TO C
5220 PRINT AT 21,(3*(I-1))+1; FLASH 1;I;
 FLASH 0
5230 NEXT I
5240 RETURN
5300 REM **CLEAR CARDS**
5310 FOR J=8 TO 21
5320 PRINT AT J,0;"
  "
5330 NEXT J
5340 RETURN
5400 REM **CLEAR CRIB**
5410 FOR J=8 TO 21
5420 PRINT AT J,21;"          "
5430 NEXT J
5440 RETURN
5500 REM **CLEAR REPORT**
5510 FOR J=1 TO 6
5520 PRINT AT J,20;"          "
5530 NEXT J
5540 GO SUB 9560
5550 RETURN
5600 REM *CLEAR BOTTOM*
5610 PRINT AT 8,0;N$
5620 RETURN
6000 REM **DRAW CARDS**
6010 FOR I=24 TO 24*CN STEP 24
6020 PLOT I,104
6030 DRAW -22,0: DRAW -2,-2: DRAW 0,-85:
 DRAW 2,-2: DRAW 22,0
6040 NEXT I
6050 DRAW 30,0: DRAW 2,2: DRAW 0,85: DRA
W -2,2: DRAW -30,0
6060 RETURN
6100 REM **PLAYERS CARDS**
6110 FOR I=C1 TO C2
```

```
6120 IF A$(I,1)="H" THEN LET Q$(I)="f":
 LET I(I)=2
6130 IF A$(I,1)="C" THEN LET Q$(I)="c":
 LET I(I)=0
6140 IF A$(I,1)="D" THEN LET Q$(I)="g":
 LET I(I)=2
6150 IF A$(I,1)="S" THEN LET Q$(I)="d":
 LET I(I)=0
6151 IF A$(I,2)="0" THEN LET R$(I)=A$(I
,3): GO TO 6160
6152 IF A$(I,3)="0" THEN LET R$(I)="h"
6153 IF A$(I,3)="1" THEN LET R$(I)="J"
6154 IF A$(I,3)="2" THEN LET R$(I)="Q"
6155 IF A$(I,3)="3" THEN LET R$(I)="K"
6160 NEXT I
6170 RETURN
6200 REM **COMPLETE CARDS**
6210 FOR I=1 TO CN
6220 INK I(I)
6230 PRINT AT 10,(3*(I-1))+1;R$(I)
6240 PRINT AT 9,(3*(I-1))+1;Q$(I)
6250 NEXT I
6260 RETURN
6300 REM **CARD OUTLINE**
6305 INK 0
6310 PLOT X1,Y1
6320 DRAW 54,0: DRAW 2,-2: DRAW 0,-85: D
RAW -2,-2: DRAW -54,0: DRAW -2,2: DRAW 0
,85: DRAW 2,2
6330 RETURN
6400 REM **SHADE PACK**
6410 PRINT AT 9,24;"jbbbbbk"
6420 FOR J=10 TO 18
6430 PRINT AT J,24;"bbbbbbb"
6440 NEXT J
6450 PRINT AT 19,24;"lllllll"
6499 RETURN
7000 REM *EVALUATOR*
7001 LET S=0
7005 LET SP=0
```

```
7010 FOR X=1 TO 5
7011 LET C(X)=VAL (H$(X,2 TO 3))
7012 NEXT X
7015 GO TO 7700
7050 REM *BONUS 1*
7051 FOR X=1 TO 5
7052 IF H$(X)<>F$(1) AND H$(X,2 TO 3)="1
1" AND H$(X,1)=F$(1,1) THEN  LET S=S+1
7053 IF C(X)>10 AND C(X)<14 THEN  LET C(
X)=10
7054 NEXT X
7100 REM *PAIRS*
7110 FOR X=1 TO 4
7120 FOR Y=X+1 TO 5
7130 IF H$(X,2 TO 3)=H$(Y,2 TO 3) THEN
LET SP=SP+2
7140 NEXT Y: NEXT X
7150 LET S=S+SP
7400 REM *FLUSH*
7405 LET SF=0
7410 IF H$(1,1)<>F$(1,1) THEN  LET Y$=H$
(1,1): LET XX=1
7411 IF H$(1,1)=F$(1,1) THEN  LET Y$=H$(
2,1): LET XX=0
7412 FOR Z=2 TO 5
7413 IF H$(Z,1)=Y$ AND H$(Z,1)<>F$(1,1)
THEN  LET XX=XX+1
7414 NEXT Z
7415 IF XX=4 THEN  LET SF=4: GO TO 7420
7418 IF H$(1,1)=H$(2,1) AND H$(1,1)=H$(3
,1) AND H$(1,1)=H$(4,1) AND H$(1,1)=H$(5
,1) THEN  LET S=S+5
7420 LET S=S+SF
7500 REM 2-CARD 15*
7510 FOR X=1 TO 4
7520 FOR Y=X+1 TO 5
7550 IF C(X)+C(Y)=15 THEN  LET S=S+2
7560 NEXT Y
7570 NEXT X
7600 REM *3-CARD 15*
```

```
7610 FOR X=1 TO 3
7620 FOR Y=X+1 TO 4
7630 FOR Z=Y+1 TO 5
7650 IF C(X)+C(Y)+C(Z)=15 THEN  LET S=S+
2
7660 NEXT Z: NEXT Y: NEXT X
7675 IF C(1)+C(2)+C(4)+C(5)=15 THEN  LET
 S=S+2
7676 IF C(2)+C(3)+C(4)+C(5)=15 THEN  LET
 S=S+2
7677 IF C(1)+C(2)+C(3)+C(5)=15 THEN  LET
 S=S+2
7678 IF C(1)+C(2)+C(3)+C(4)=15 THEN  LET
 S=S+2
7679 IF C(1)+C(2)+C(3)+C(4)+C(5)=15 THEN
  LET S=S+2
7680 GO TO 7900
7700 IF C(1)+1=C(2) AND C(2)+1=C(3) AND
C(3)+1=C(4) AND C(4)+1=C(5) THEN  LET S=
S+5: GO TO 7748
7701 LET R4=0
7702 FOR W=1 TO 2
7703 FOR X=W+1 TO 3
7704 FOR Y=X+1 TO 4
7705 FOR Z=Y+1 TO 5
7706 IF C(W)=C(X)-1 AND C(X)=C(Y)-1 AND
C(Y)=C(Z)-1 THEN  LET R4=R4+4
7707 NEXT Z: NEXT Y: NEXT X: NEXT W
7708 IF R4<>0 THEN  LET S=S+R4: GO TO 77
48
7710 FOR X=1 TO 3
7720 FOR Y=X+1 TO 4
7730 FOR Z=Y+1 TO 5
7746 IF C(Y)-C(X)=1 AND C(Z)-C(Y)=1 THEN
  LET S=S+3
7747 NEXT Z: NEXT Y: NEXT X
7748 GO TO 7050
7900 IF S>MAX THEN  LET MAX=S: LET N1=I:
 LET N2=J: LET N3=K: LET N4=L
7999 RETURN
```

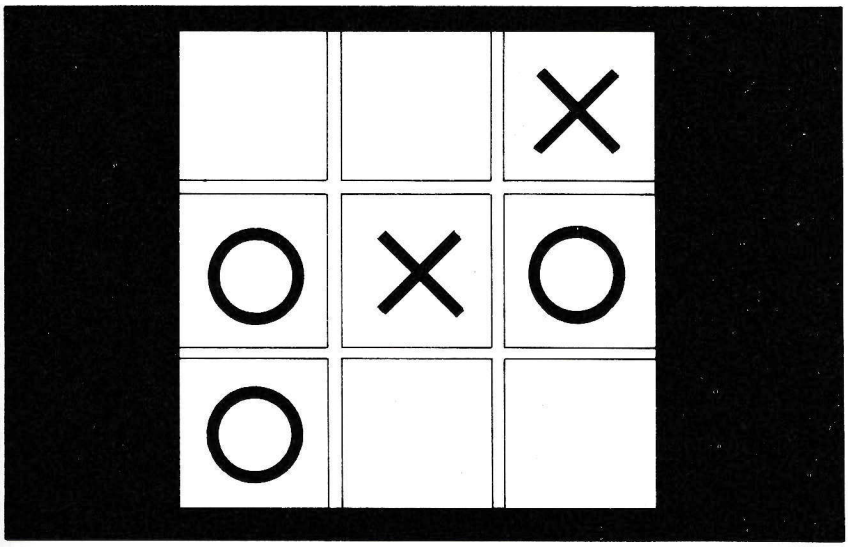
```
8000 REM *HAND SELECTION*
8001 LET H$(5)="999"
8005 LET MAX=0
8010 FOR I=1 TO 3
8020 FOR J=I+1 TO 4
8030 FOR K=J+1 TO 5
8040 FOR L=K+1 TO 6
8050 LET H$(1)=A$(I)
8060 LET H$(2)=A$(J)
8070 LET H$(3)=A$(K)
8080 LET H$(4)=A$(L)
8085 GO SUB 7000
8090 NEXT L: NEXT K: NEXT J: NEXT I
8095 IF MAX=0 THEN  LET N1=1: LET N2=2:
LET N3=3: LET N4=4
8100 LET I$(1)=A$(N1): LET I$(2)=A$(N2):
 LET I$(3)=A$(N3): LET I$(4)=A$(N4)
8110 LET K=1
8111 FOR I=1 TO 6
8112 FOR J=1 TO 4
8114 IF A$(I)=I$(J) THEN  GO TO 8118
8115 NEXT J
8116 LET C$(K)=A$(I): LET K=K+1
8118 NEXT I
8120 RETURN
8300 REM **GAME OVER**
8310 FOR I=1 TO 4
8320 FOR J=0 TO 7
8330 BORDER J
8340 NEXT J
8350 NEXT I
8360 GO SUB 5500
8370 PRINT AT 1,21;"GAME OVER"
8380 PRINT AT 2,21;Z$
8390 PRINT AT 3,21;"WINS."
8410 PRINT AT 4,21;"PRESS ENTER"
8420 PRINT AT 5,21;;"TO CONTINUE"
8430 GO SUB 9560
8440 LET Z$=INKEY$
8450 IF Z$="" THEN  GO TO 8440
8460 RUN
9000 REM *CARDS*
9010 LET S$="HCDS"
```

```
9020 LET V$="0102030405060708091011121 3"
9030 FOR I=1 TO 12
9040 LET A=INT (RND*4)+1
9050 LET A$(I,1)=S$(A)
9060 LET A=INT (RND*26)+1
9070 IF A/2=INT (A/2) THEN  GO TO 9060
9080 LET A$(I,2 TO 3)=V$(A TO A+1)
9090 FOR J=1 TO I-1
9100 IF A$(J)=A$(I) THEN  GO TO 9040
9110 NEXT J
9120 NEXT I
9125 RETURN
9130 FOR I=S TO F
9140 LET B$(I)=A$(I)
9150 NEXT I
9155 FOR J=S TO F
9160 LET MIN=99
9170 FOR I=S TO F
9180 IF VAL (B$(I,2 TO 3))<MIN THEN  GO
SUB 9300
9190 NEXT I
9200 LET A$(J)=B$(K)
9205 LET B$(K,2 TO 3)="99"
9210 NEXT J
9220 RETURN
9300 LET MIN=VAL (B$(I,2 TO 3))
9310 LET K=I
9320 RETURN
9400 REM **TEMPORARY TRANSFER**
9410 FOR I=1 TO 12
9420 LET B$(I)=A$(I)
9430 NEXT I
9440 RETURN
9500 REM *CRIB-BOARD*
9510 FOR I=1 TO 30
9520 PLOT I*5,165: PLOT I*5,155: PLOT I*
5,139: PLOT I*5,129
9530 NEXT I
9540 PLOT 6,147
9550 PLOT 0,175: DRAW 156,0: DRAW 0,-55:
 DRAW -156,0: DRAW 0,55
9560 PLOT 160,175: DRAW 95,0: DRAW 0,-55
: DRAW -95,0: DRAW 0,55
9699 RETURN
```

```
9900 REM *UDG'S*
9910 FOR I=USR "A" TO USR "M"+7
9920 READ A
9930 POKE I,A
9935 NEXT I
9936 RETURN
9940 DATA 0,0,0,0,0,0,0,0
9950 DATA 170,85,170,85,170,85,170,85
9960 DATA 0,56,56,16,214,254,214,16
9970 DATA 0,16,56,124,254,254,214,16
9980 DATA 170,170,85,85,170,170,85,85
9990 DATA 0,108,254,254,124,124,56,16
9991 DATA 0,16,56,124,254,254,124,16
9992 DATA 152,164,164,164,164,164,164,15
2
9993 DATA 128,128,128,128,128,128,128,12
8
9994 DATA 42,85,170,85,170,85,170,85
9995 DATA 169,84,170,85,170,85,170,85
9996 DATA 170,85,170,85,170,85,170,21
9997 DATA 170,85,170,85,170,85,170,85
9999 IF C$(2)=C$(3) AND C$(3)<>"    " THE
N  GO TO 51
```

Chapter 7

OXO



Description

The game of noughts and crosses is a very old one suitable for two players, and learning how to play it is relatively simple. In this version, the two players involved are you and the computer, and you each have a symbol—either a nought (0) or a cross (X). You each take turns at placing your symbol in one of the nine squares on a 3 × 3 grid, trying to form a straight line of your symbol either horizontally, vertically or diagonally.

The game ends when one player completes the row of symbols, in which case that player is the winner, or when all the squares are occupied but with no line of symbols. In the latter case, the game is drawn.

Hints on Entry

Care must be taken when entering lines 300–440 and with the subroutine starting at line 1700. These lines form the brain of the program.

In lines 300–440, the routine which the computer uses to decide to allow the machine to have the first move is defined, and the subroutine is where the computer discovers if there is the possibility of obtaining or blocking a winning line. Therefore a mistake in either of these sections could lead to ridiculous moves, with the computer selecting or permitting you to occupy squares which have already been occupied.

When you have entered the program, save it on tape or microdrive with the command:

```
SAVE "0X0" LINE 1
```

or

```
SAVE *"M";1;"0X0" LINE 1
```

Techniques

On occasions we may wish to print blocks of colour onto the screen very quickly. This can be achieved by poking directly into the memory locations reserved for the display file.

Unlike most computers, the Sinclair Spectrum does not employ a simple block of memory through which it is possible to poke characters directly onto the screen, however there is a block reserved for the screen attributes (colours, flash status, bright status, etc.) and it is possible to poke values directly into this portion of memory, producing some interesting effects on the screen.

Example

```
10 CLS
20 FOR I = 1 TO 100
30 LET X = INT(RND*8)
40 LET Y = INT(RND*768)
50 POKE 22528+Y,X*8
60 NEXT I
```

This program produces squares of random colour, at random positions on the screen. The value of X gives the colour and Y the number representing the position on the screen. The area of memory reserved for information concerning these attributes commences at 22528, and therefore by adding the value of Y to this we obtain the memory location representing the appropriate screen location.

If we need to put a block of colour on the screen at a particular location (say X,Y), then we must first calculate the appropriate memory location representing the screen position. This is achieved by using the formula:

POSITION = 22528 + X + (Y*32)

Then we can place the block of colour at this position by using POKE POSITION, C where C is eight times the respective colour number required.

Instructions

When the program is executed, a brief résumé of the object of noughts and crosses is displayed on the screen before you are given the choice of having the first move or allowing the computer to go first. If you decide to go first, you then have to select the level of the play with level one being for novices and level five for experts. If you allow the computer to go first then it will play to win.

Your symbol is the 0, with the computer playing X. When the computer prompts you to take your turn, press the key corresponding to the number in the square which you wish to occupy. Your symbol will then appear and the Spectrum will take its move.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED BY LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
  1 POKE 23658,8
  5 DIM B(9): LET A=0: LET DRAW=0
  6 BORDER 1
 10 GO SUB 1100
 15 PRINT : PRINT : INK 0
 20 PRINT "THE OBJECT OF THIS GAME IS T
O BETHE FIRST TO OBTAIN A ROW OF    SYMB
OLS ON THE 3X3 GRID."
 21 PRINT
 25 PRINT "YOU HAVE THE CHOICE OF GOING
   FIRST OR ALLOWING THE COMPUTER  TO H
AVE THE FIRST MOVE."
 26 PRINT
 30 PRINT "INDICATE WHICH SQUARE YOU WI
SH  TO OCCUPY BY PRESSING THE       APPR
OPRIATE KEY IN THE RANGE    0 TO 9."
 31 PRINT : PRINT TAB (9);: FLASH 1: PR
INT "PRESS ANY KEY": FLASH 0
 32 IF INKEY$="" THEN  GO TO 32
 33 CLS
 34 GO SUB 1100: INK 0
 35 PRINT : PRINT : PRINT "YOU ALSO HAV
E A CHOICE OF FIVE  LEVELS OF PLAY."
 36 PRINT : PRINT "LEVEL 1 IS FOR NOVIC
ES AND LEVEL5 IS FOR EXPERTS, BUT BE WAR
NED,IF YOU LET THE COMPUTER GO FIRSTIT W
ILL PLAY TO WIN!!"
 37 PRINT AT 21,9;"";: FLASH 1: PRINT "
PRESS ANY KEY";: FLASH 0
 38 IF INKEY$="" THEN  GO TO 38
 39 CLS
 44 GO SUB 1100
 49 INK 0: PRINT
 50 PRINT " DO YOU WANT TO GO FIRST? (Y
/N)"
 55 PAUSE 10: LET F$=INKEY$: IF F$="" T
HEN  GO TO 55
 56 IF F$<>"Y" AND F$<>"N" THEN  GO TO
55
```

```
  60 IF F$="N" THEN  GO TO 65
  61 PRINT : PRINT " ENTER LEVEL (1-EASY
 TO 5-HARD)"
  62 LET L$=INKEY$: IF CODE (L$)<49 OR C
ODE (L$)>53 THEN  GO TO 62
  63 LET L=.5+VAL (L$)/10
  65 PRINT AT 11,3;" YOU ARE PLAYING NOU
GHTS"
  66 LET X=12: LET Y=14 : GO SUB 1300
  67 INK 0: PRINT AT 17,1;"THE COMPUTER
IS PLAYING CROSSES"
  68 LET X=18: LET Y=14: GO SUB 1200
  70 PAUSE 1: PAUSE 200
  75 CLS : GO SUB 1100: GO SUB 1000
  80 FOR I=1 TO 5
  85 LET X=4+3*I: LET Y=3: GO SUB 1300
  90 LET Y=25: GO SUB 1200
  95 NEXT I
 100 IF F$="N" THEN  GO TO 300
 101 INK 0
 105 GO SUB 2000
 125 LET X=19: LET Y=3: INK 7: GO SUB 13
05: INK 0
 135 IF S<>5 THEN  LET S=5: GO TO 145
 140 LET S=1+2*INT (5*RND): IF S=5 THEN
 GO TO 140
 142 PAUSE 1: PAUSE 50
 145 GO SUB 2100: GO SUB 1400: GO SUB 12
00: LET X=19: LET Y=25: INK 7: GO SUB 12
05: INK 0: LET B(S)=2
 150 FOR A=1 TO 3
 155 GO SUB 2000
 160 LET X=19-3*A: LET Y=3: INK 7: GO SU
B 1305:: INK 0
 177 GO SUB 1500: IF T=1 THEN  GO TO 190
0
 190 LET K=2: GO SUB 1700
 195 IF S<>0 THEN  GO SUB 2100: LET B(S)
=2: GO SUB 1400: GO SUB 1200: GO TO 240
```

```
 200 LET K=1: GO SUB 1700
 205 IF S<>0 THEN  GO SUB 2100: LET B(S)
=2: GO SUB 1400: GO SUB 1200: GO TO 240
 210 LET S=1+INT (9*RND)
 215 IF B(S)=0 THEN  GO TO 225
 220 GO TO 210
 222 GO SUB 2100
 225 LET B(S)=2: GO SUB 1400: GO SUB 120
0
 240 LET X=19-A*3: LET Y=25: INK 7: GO S
UB 1205: INK 0
 245 GO SUB 1500
 250 IF T=1 THEN  GO TO 1900
 255 NEXT A
 270 NEXT A
 275 GO SUB 2000: GO SUB 1400: GO SUB 13
00
 280 LET X=7: LET Y=3: INK 7: GO SUB 130
5: INK 0
 285 GO SUB 1500
 290 IF T=0 THEN  LET DRAW=1
 295 GO TO 1900
 299 STOP
 300 PAUSE 50: LET S=1+INT (9*RND): IF S
/2=INT (S/2) THEN  GO TO 300
 302 LET B(S)=2
 310 GO SUB 1400: GO SUB 1200: LET X=19:
 LET Y=25: INK 7: GO SUB 1205: INK 0
 312 IF S<>5 THEN  GO TO 400
 315 GO SUB 2000
 317 LET X=19: LET Y=3: INK 7: GO SUB 13
05: INK 0
 320 IF S/2<>INT (S/2) THEN  GO TO 350
 321 IF S=2 OR S=4 THEN  LET S=9
 322 IF S=6 OR S=8 THEN  LET S=1
 323 LET B(S)=2
 324 GO SUB 1400: GO SUB 1200: LET X=16:
 LET Y=25: INK 7: GO SUB 1205: INK 0
 326 GO SUB 2000
```

```
 328 LET X=16: LET Y=3: INK 7: GO SUB 13
05: INK 0
 330 LET K=2: GO SUB 1700: IF S<>0 THEN
 GO TO 333
 331 LET K=1: GO SUB 1700: IF S<>0 THEN
 GO TO 333
 334 GO SUB 1400: GO SUB 1200: LET B(S)=
2: LET X=13: LET Y=25: INK 7: GO SUB 120
5: INK 0
 336 GO SUB 1500: IF T=1 THEN  GO TO 190
0
 338 GO SUB 2000: LET X=13: LET Y=3: INK
 7: GO SUB 1305: INK 0
 340 LET K=2: GO SUB 1700: LET B(S)=2: G
O SUB 1400: GO SUB 1200
 342 LET X=10: LET Y=25: INK 7: GO SUB 1
205: INK 0
 346 GO SUB 1900
 350 LET S=10-S: LET S1=S: LET B(S)=2: G
O SUB 1400: GO SUB 1200
 352 LET X=16: LET Y=25: INK 7: GO SUB 1
205: INK 0
 354 GO SUB 2000
 355 LET X=16: LET Y=3: INK 7: GO SUB 13
05: INK 0
 356 LET K=1: GO SUB 1700: IF S<>0 THEN
 GO TO 375
 357 IF S1<>9 THEN  GO TO 360
 358 IF B(6)=1 THEN  LET S=8
 359 IF B(8)=1 THEN  LET S=6
 360 IF S1<>7 THEN  GO TO 363
 361 IF B(4)=1 THEN  LET S=8
 362 IF B(8)=1 THEN  LET S=4
 363 IF S1<>3 THEN  GO TO 366
 364 IF B(2)=1 THEN  LET S=6
 365 IF B(6)=1 THEN  LET S=2
 366 IF S1<>1 THEN  GO TO 369
 367 IF B(2)=1 THEN  LET S=4
 368 IF B(4)=1 THEN  LET S=2
```

```
 369 LET B(S)=2: GO SUB 1400: GO SUB 120
0: LET X=13: LET Y=25: INK 7: GO SUB 120
5: INK 0
 370 GO SUB 2000: LET X=13: LET Y=3: INK
 7: GO SUB 1305: INK 0
 371 LET K=2: GO SUB 1700: LET B(S)=2: G
O SUB 1400: GO SUB 1200
 372 LET X=10: LET Y=25: INK 7: GO SUB 1
205: INK 0
 373 GO SUB 1500: GO SUB 1900
 375 LET K=1: GO SUB 1700: LET B(S)=2: G
O SUB 1400: GO SUB 1200: LET X=13: LET Y
=25: INK 7: GO SUB 1205: INK 0
 376 GO SUB 2000: LET X=13: LET Y=3: INK
 7: GO SUB 1305: INK 0
 377 LET K=2: GO SUB 1700: IF S<>0 THEN
 GO TO 380
 378 LET K=1: GO SUB 1700: IF S<>0 THEN
 GO TO 380
 379 LET S=1+INT (9*RND): IF B(S)<>0 THE
N  GO TO 379
 380 LET B(S)=2: GO SUB 1400: GO SUB 120
0: LET X=10: LET Y=25: INK 7: GO SUB 120
5: INK 0: GO SUB 1500: IF T=1 THEN  GO T
O 1900
 382 GO SUB 2000: LET X=10: LET Y=3: INK
 7: GO SUB 1305: INK 0
 384 FOR A=1 TO 9: IF B(A)=0 THEN  LET S
=A
 385 NEXT A
 386 LET B(S)=2: GO SUB 1400: GO SUB 120
0: LET X=7: LET Y=25: INK 7: GO SUB 1205
: INK 0: LET DRAW=1: GO TO 1900
 400 LET S1=S: GO SUB 2000: LET X=19: LE
T Y=3: INK 7: GO SUB 1305: INK 0
 405 IF S<>5 THEN  LET S=5: GO TO 415
 410 LET S=10-S1
 415 LET B(S)=2: GO SUB 1400: GO SUB 120
0: LET X=16: LET Y=25: INK 7: GO SUB 120
5: INK 0
```

```
 419 FOR B=1 TO 3
 420 GO SUB 2000: LET X=19-3*B: LET Y=3:
 INK 7: GO SUB 1305: INK 0
 425 LET K=2: GO SUB 1700: IF S<>0 THEN
 GO TO 430
 426 LET K=1: GO SUB 1700: IF S<>0 THEN
 GO TO 430
 427 IF B=1 AND B(5)=2 THEN  GO TO 357
 429 LET S=1+INT (9*RND): IF B(S)<>0 THE
N  GO TO 426
 430 LET B(S)=2: GO SUB 1400: GO SUB 120
0: LET X=16-3*B: LET Y=25: INK 7: GO SUB
 1205: INK 0
 435 GO SUB 1500: IF T=1 THEN  GO TO 190
0
 436 NEXT B
 440 LET DRAW=1: GO TO 1900
 998 INK 0
 999 STOP
1000 INK 3
1005 FOR I=0 TO 10
1010 PRINT AT 7+I,13;CHR$ 143
1020 PRINT AT 7+I,17;CHR$ 143
1030 PRINT AT 10,10+I;CHR$ 143
1040 PRINT AT 14,10+I;CHR$ 143
1050 NEXT I
1055 INK 0
1060 FOR I=1 TO 9
1065 IF I<4 THEN  PRINT AT 8,7+4*I;I: GO
 TO 1080
1070 IF I>6 THEN  PRINT AT 16,7+4*(I-6);
I: GO TO 1080
1075 PRINT AT 12,7+4*(I-3);I
1080 NEXT I
1090 RETURN
1100 INK 2
1104 PRINT
1105 PRINT TAB (5);"**********************
*"
1110 PRINT TAB (5);"*";TAB (25);"*"
```

```
1120 PRINT TAB (5);"* NOUGHTS & CROSSES
*"
1130 PRINT TAB (5);"*";TAB (25);"*"
1140 PRINT TAB (5);"**********************
*"
1150 RETURN
1200 INK 5
1205 PRINT AT X,Y;CHR$ 132
1210 PRINT AT X,Y+2;CHR$ 136
1220 PRINT AT X+1,Y+1;CHR$ 143
1230 PRINT AT X+2,Y;CHR$ 129
1240 PRINT AT X+2,Y+2;CHR$ 130
1250 RETURN
1300 INK 4
1305 PRINT AT X,Y;CHR$ 132
1310 PRINT AT X,Y+2;CHR$ 136
1315 PRINT AT X,Y+1;CHR$ 140
1320 PRINT AT X+1,Y;CHR$ 133
1325 PRINT AT X+1,Y+2;CHR$ 138
1330 PRINT AT X+2,Y;CHR$ 129
1335 PRINT AT X+2,Y+1;CHR$ 131
1340 PRINT AT X+2,Y+2;CHR$ 130
1345 PRINT AT X+1,Y+1;CHR$ 128
1350 RETURN
1400 LET X=7+4*INT ((S-1)/3)
1410 LET Y=6+4*(S-3*INT ((S-1)/3))
1420 RETURN
1500 LET DRAW=0: LET T=0
1505 FOR J=1 TO 2
1510 IF B(1)=J AND B(2)=J AND B(3)=J THE
N  GO TO 1600
1515 IF B(1)=J AND B(5)=J AND B(9)=J THE
N  GO TO 1600
1520 IF B(1)=J AND B(4)=J AND B(7)=J THE
N  GO TO 1600
1525 IF B(2)=J AND B(5)=J AND B(8)=J THE
N  GO TO 1600
1530 IF B(3)=J AND B(6)=J AND B(9)=J THE
N  GO TO 1600
```

```
1535 IF B(3)=J AND B(5)=J AND B(7)=J THE
N  GO TO 1600
1540 IF B(4)=J AND B(5)=J AND B(6)=J THE
N  GO TO 1600
1545 IF B(7)=J AND B(8)=J AND B(9)=J THE
N  GO TO 1600
1550 NEXT J
1555 RETURN
1600 LET T=1: RETURN
1700 LET S=0
1701 IF B(1)=1 AND B(9)=1 AND B(5)=2 AND
 A=1 THEN  LET S=2: GO TO 1800
1702 IF B(3)=1 AND B(7)=1 AND B(5)=2 AND
 A=1 THEN  LET S=8: GO TO 1800
1703 IF B(1)=1 AND B(6)=1 AND B(5)=2 AND
 A=1 THEN  LET S=9: GO TO 1800
1704 IF B(1)=1 AND B(8)=1 AND B(5)=2 AND
 A=1 THEN  LET S=9: GO TO 1800
1705 IF B(1)=K AND B(2)=K AND B(3)=0 THE
N  LET S=3: GO TO 1800
1706 IF B(2)=1 AND B(4)=1 AND B(5)=2 AND
 A=1 THEN  LET S=1: GO TO 1800
1707 IF B(2)=1 AND B(6)=1 AND B(5)=2 AND
 A=1 THEN  LET S=3: GO TO 1800
1708 IF B(8)=1 AND B(6)=1 AND B(5)=2 AND
 A=1 THEN  LET S=9: GO TO 1800
1709 IF B(8)=1 AND B(4)=1 AND B(5)=2 AND
 A=1 THEN  LET S=7: GO TO 1800
1710 IF B(1)=K AND B(3)=K AND B(2)=0 THE
N  LET S=2: GO TO 1800
1715 IF B(1)=K AND B(4)=K AND B(7)=0 THE
N  LET S=7: GO TO 1800
1716 IF A=1 AND B(5)=1 AND B(9)=1 AND B(
1)=2 AND B(7)=0 THEN  LET S=7: GO TO 180
0
1717 IF A=1 AND B(5)=1 AND B(7)=1 AND B(
3)=2 AND B(9)=0 THEN  LET S=9: GO TO 180
0
1718 IF A=1 AND B(5)=1 AND B(3)=1 AND B(
```

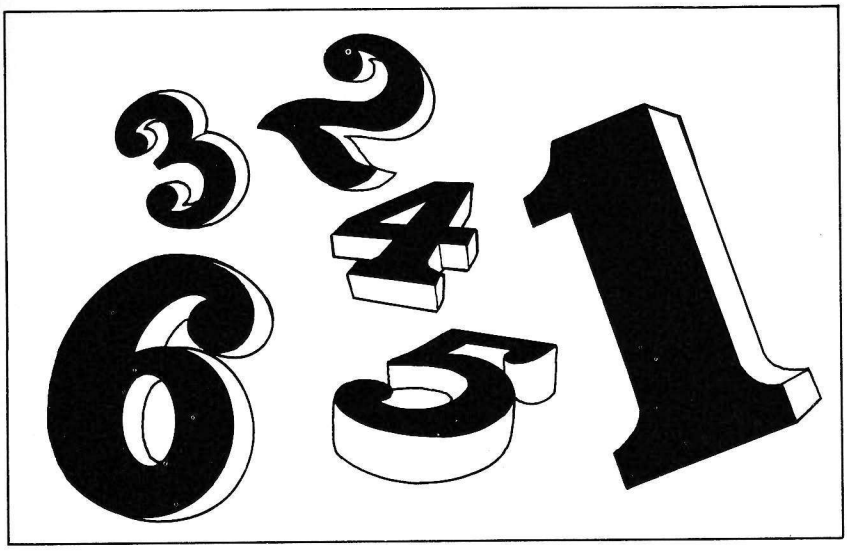
```
7)=2 AND B(1)=0 THEN  LET S=1: GO TO 180
0
1719 IF A=1 AND B(5)=1 AND B(1)=1 AND B(
9)=2 AND B(3)=0 THEN  LET S=3: GO TO 180
0
1720 IF B(1)=K AND B(5)=K AND B(9)=0 THE
N  LET S=9: GO TO 1800
1725 IF B(1)=K AND B(7)=K AND B(4)=0 THE
N  LET S=4: GO TO 1800
1730 IF B(1)=K AND B(9)=K AND B(5)=0 THE
N  LET S=5: GO TO 1800
1731 IF B(3)=1 AND B(4)=1 AND B(5)=2 AND
 A=1 THEN  LET S=7: GO TO 1800
1732 IF B(3)=1 AND B(8)=1 AND B(5)=2 AND
 A=1 THEN  LET S=7: GO TO 1800
1735 IF B(2)=K AND B(3)=K AND B(1)=0 THE
N  LET S=1: GO TO 1800
1740 IF B(2)=K AND B(5)=K AND B(8)=0 THE
N  LET S=8: GO TO 1800
1741 IF B(7)=1 AND B(2)=1 AND B(5)=2 AND
 A=1 THEN  LET S=3: GO TO 1800
1742 IF B(7)=1 AND B(6)=1 AND B(5)=2 AND
 A=1 THEN  LET S=3: GO TO 1800
1745 IF B(2)=K AND B(8)=K AND B(5)=0 THE
N  LET S=5: GO TO 1800
1750 IF B(3)=K AND B(5)=K AND B(7)=0 THE
N  LET S=7: GO TO 1800
1751 IF B(9)=1 AND B(2)=1 AND B(5)=2 AND
 A=1 THEN  LET S=1: GO TO 1800
1752 IF B(9)=1 AND B(4)=1 AND B(5)=2 AND
 A=1 THEN  LET S=1: GO TO 1800
1755 IF B(3)=K AND B(6)=K AND B(9)=0 THE
N  LET S=9: GO TO 1800
1760 IF B(3)=K AND B(7)=K AND B(5)=0 THE
N  LET S=5: GO TO 1800
1763 IF B(3)=K AND B(9)=K AND B(6)=0 THE
N  LET S=6: GO TO 1800
1766 IF B(4)=K AND B(5)=K AND B(6)=0 THE
N  LET S=6: GO TO 1800
1769 IF B(4)=K AND B(6)=K AND B(5)=0 THE
N  LET S=5: GO TO 1800
```

```
1772 IF B(4)=K AND B(7)=K AND B(1)=0 THE
N  LET S=1: GO TO 1800
1775 IF B(6)=K AND B(5)=K AND B(4)=0 THE
N  LET S=4: GO TO 1800
1778 IF B(6)=K AND B(9)=K AND B(3)=0 THE
N  LET S=3: GO TO 1800
1780 IF B(7)=K AND B(5)=K AND B(3)=0 THE
N  LET S=3: GO TO 1800
1781 IF B(7)=K AND B(8)=K AND B(9)=0 THE
N  LET S=9: GO TO 1800
1784 IF B(7)=K AND B(9)=K AND B(8)=0 THE
N  LET S=8: GO TO 1800
1787 IF B(8)=K AND B(9)=K AND B(7)=0 THE
N  LET S=7: GO TO 1800
1788 IF B(8)=K AND B(5)=K AND B(2)=0 THE
N  LET S=2: GO TO 1800
1790 IF B(9)=K AND B(5)=K AND B(1)=0 THE
N  LET S=1: GO TO 1800
1800 RETURN
1900 IF DRAW=1 THEN  PRINT AT 19,10;"MAT
CH DRAWN": GO TO 1930
1910 IF J=1 THEN  PRINT AT 19,10;"NOUGHT
S WINS": GO TO 1930
1920 PRINT AT 19,10;"CROSSES WINS"
1930 PAUSE 1: PAUSE 100
1940 PAUSE 10: PRINT AT 19,8;"ANOTHER GO
 (Y/N)"
1950 LET Z$=INKEY$: IF Z$="" THEN  GO TO
 1950
1960 IF Z$="N" THEN  STOP
1965 FOR I=1 TO 9: LET B(I)=0: NEXT I
1970 CLS : GO TO 40
2000 FLASH 1: PRINT AT 19,11;"YOUR MOVE"
: FLASH 0
2010 LET S$=INKEY$: IF S$="" THEN  GO TO
 2010
2020 IF CODE (S$)<49 OR CODE (S$)>57 THE
N  GO TO 2000
2022 IF B(VAL (S$))<>0 THEN  GO TO 2010
2024 LET S=VAL (S$)
```

```
2025 LET B(S)=1
2028 GO SUB 1400: GO SUB 1300
2029 PRINT AT 19,11;"          "
2030 RETURN
2100 IF RND<L THEN  RETURN
2110 LET S=1+INT (9*RND): IF B(S)<>0 THE
N  GO TO 2110
2120 RETURN
```

Chapter 8

Simon



Description

How many times have you excused yourself by saying '. . . sorry, it just slipped my memory', when you have forgotten that vital piece of information? Well, all you need to succeed at Simon is a very good memory.

The object of the game is to reproduce the order in which the computer flashes a sequence of numbers in the range one to six. But be warned—the longer the sequence of numbers, the quicker they are displayed.

Hints on Entry

The program is fairly short and should not give any problem when being entered. The only part of the program which must be entered with special care is the subroutine at line 1000. This is the routine which places into the computer's memory the machine code program to make the screen flash, and consequently the data at line 1040 must be input accurately.

As with all programs involving machine code routines, the program should be saved on tape or microdrive before being executed and the command to do this is:

```
SAVE "SIMON" LINE 1
```

or

```
SAVE *"M";1;"SIMON" LINE 1
```

Techniques

In Chapter 25 of the Spectrum manual there is a section on what are known as system variables. For the beginner to computing, the concepts involved are extremely complex as one needs to understand something about machine code and the internal workings of the machine to understand their meaning and usefulness. However, by manipulating these variables using the POKE command, some interesting and useful facilities can be made available. A few of these are described below.

POKE 23562,X	Alters the speed at which a key automatically repeats. The smaller the value of X the faster the repeat.
POKE 23609,X	Alters the duration of the click obtained when a key is depressed.
POKE 23617,X	Changes the cursor from one mode to another.
POKE 23624,X	Changes the border colour very quickly. The value of X must be eight times the colour required.
POKE 23659,X	Specifies the number of lines in the lower part of the screen. It can be used to divide the screen into two distinct windows.

POKE 23692,X Specifies the number of lines which will be displayed in the top part of the screen before the SCROLL? message appears.

These are just a few of the most useful system variables which can be altered very easily, thus changing the way in which the machine operates. In each case, it must be remembered that the value of X must be in the range 0–255.

Instructions

The program is automatically executed when it has been loaded. After a brief explanation of the game, the computer decides on the random sequence and six squares are displayed on the screen. Each of these squares has a number in it, and it is this number which you must press when it is your turn to reproduce the computer's moves.

Each time you correctly echo the computer's sequence, the sequence is increased by one and repeated for you to try again. The game ends as soon as you make a mistake, and if your score is a new record then you are asked to input your name (with a maximum of eight characters) which is then displayed for all to see on the right-hand side of the screen.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
 4 POKE 23658,8
 5 CLEAR 60449
 6 RANDOMIZE
10 DIM N(50)
15 BORDER 1: INK 1
20 GO SUB 800
25 LET REC=10
```

```
  30 LET B$="SPECTRUM"
  45 PRINT : PRINT
  50 PRINT "THE OBJECT OF THE GAME SIMON
 IS TO REPRODUCE A GIVEN SEQUENCE OFEVEN
TS."
  55 PRINT
  60 PRINT "THE SCREEN WILL SHOW SIX SQU
ARESAND THESE WILL FLASH IN A RANDOMORDE
R."
  65 PRINT "YOUR TASK IS TO REPRODUCE TH
E   ORDER IN WHICH THIS OCCURS BY   PRES
SING THE CORRESPONDING KEYS IN THE RANGE
 1 TO 6."
  70 PRINT
  75 PRINT "PRESS ANY KEY TO CONTINUE"
  80 GO SUB 1000
  85 GO SUB 900
  90 IF INKEY$="" THEN  GO TO 90
 100 CLS
 105 GO SUB 800
 110 LET P=22734: LET C=1: GO SUB 1100
 115 LET P=22836: LET C=2: GO SUB 1100
 120 LET P=22964: LET C=3: GO SUB 1100
 125 LET P=23054: LET C=4: GO SUB 1100
 130 LET P=22952: LET C=5: GO SUB 1100
 135 LET P=22824: LET C=6: GO SUB 1100
 140 PRINT AT 7,15;"1"
 145 PRINT AT 10,21;"2"
 150 PRINT AT 14,21;"3"
 155 PRINT AT 17,15;"4"
 160 PRINT AT 14,9;"5"
 165 PRINT AT 10,9;"6"
 170 PRINT AT 10,1;"LENGTH"
 175 PRINT AT 12,4;"0"
 180 PRINT AT 10,24;"RECORD"
 185 PRINT AT 12,26;REC
 190 PRINT AT 14,26;"BY"
 195 PRINT AT 16,24;B$
 200 FOR J=1 TO 50
```

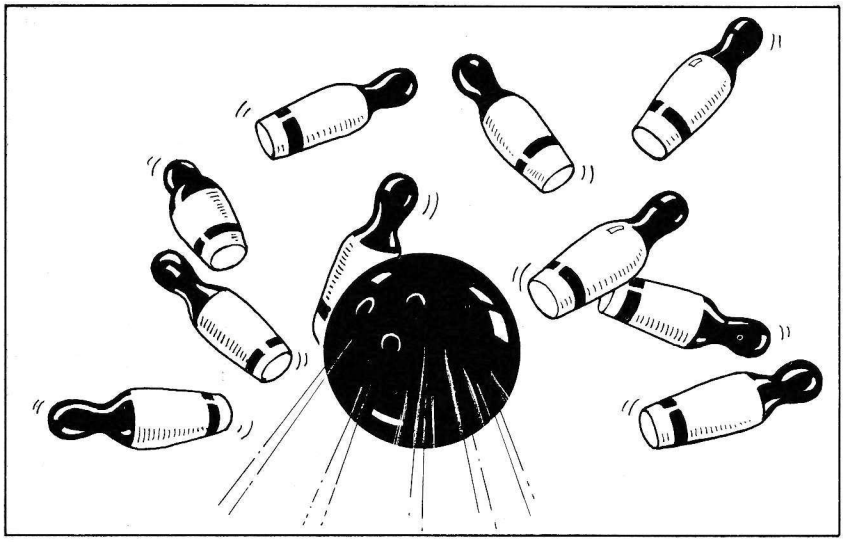
```
 205 PAUSE 25
 210 FOR I=1 TO J
 215 IF J<10 THEN  PAUSE 25-2.5*J
 220 GO SUB N(I)*50+450
 230 NEXT I
 240 PRINT AT 20,11;"YOUR TURN"
 250 FOR I=1 TO J
 260 LET N$=INKEY$: IF N$="" THEN  GO TO
 260
 270 IF CODE (N$)<49 OR CODE (N$)>54 THE
N  GO TO 260
 280 IF VAL (N$)<>N(I) THEN  GO TO 1200
 290 GO SUB N(I)*50+450
 300 NEXT I
 310 PRINT AT 20,11;"          "
 320 IF J<10 THEN  PRINT AT 12,4;J
 330 IF J>=10 THEN  PRINT AT 12,3;J
 340 IF J>REC THEN  PRINT AT 12,26;J: PR
INT AT 16,24;"          "
 490 NEXT J
 499 STOP
 500 LET C=7: LET P=22734: GO SUB 1100
 505 BEEP .2,7
 510 LET C=1: LET P=22734: GO SUB 1100
 515 PRINT AT 7,15;"1"
 520 RETURN
 550 LET C=7: LET P=22836: GO SUB 1100
 555 BEEP .2,9
 560 LET C=2: LET P=22836: GO SUB 1100
 565 PRINT AT 10,21;"2"
 570 RETURN
 600 LET C=7: LET P=22964: GO SUB 1100
 605 BEEP .2,11
 610 LET C=3: LET P=22964: GO SUB 1100
 615 PRINT AT 14,21;"3"
 620 RETURN
 650 LET C=7: LET P=23054: GO SUB 1100
 655 BEEP .2,12
 660 LET C=4: LET P=23054: GO SUB 1100
```

```
665 PRINT AT 17,15;"4"
670 RETURN
700 LET C=7: LET P=22952: GO SUB 1100
705 BEEP .2,14
710 LET C=5: LET P=22952: GO SUB 1100
715 PRINT AT 14,9;"5"
720 RETURN
750 LET C=7: LET P=22824: GO SUB 1100
755 BEEP .2,16
760 LET C=6: LET P=22824: GO SUB 1100
765 PRINT AT 10,9;"6"
770 RETURN
799 STOP
800 PRINT TAB (9);"**********"
810 PRINT TAB (9);"*";TAB (21);"*"
820 PRINT TAB (9);"* S I M O N *"
830 PRINT TAB (9);"*";TAB (21);"*"
840 PRINT TAB (9);"**********"
850 RETURN
900 FOR I=1 TO 50
910 LET N(I)=1+INT (RND*6)
920 NEXT I
930 RETURN
1000 FOR I=60453 TO 60477
1010 READ A
1020 POKE I,A
1030 NEXT I
1035 RETURN
1040 DATA 42,34,236,58,36,236,17,29,0,14
,3,6,3,119,35,5,194,50,236,25,13,194,48,
236,201
1100 LET H=INT (P/256)
1110 LET L=P-H*256
1120 POKE 60450,L
1130 POKE 60451,H
1140 POKE 60452,7+7*C
1150 LET ZZ=USR 60453
1160 RETURN
1200 PRINT AT 12,11;"GAME OVER"
```

```
1205 GO SUB 1400
1210 PAUSE 250
1215 LET J=J-1
1220 IF J<=REC THEN  GO TO 1250
1225 PRINT AT 20,8;"ENTER YOUR NAME";
1227 LET B$=""
1230 FOR K=24 TO 31
1232 PAUSE 0: LET Z$=INKEY$: LET B$=B$+Z
$
1234 PRINT AT 16,K;Z$
1236 IF CODE (Z$)=13 THEN  GO TO 1245
1240 NEXT K
1245 LET REC=J
1250 PRINT AT 20,6;"   ANOTHER GO ?
  "
1260 PAUSE 0: LET Z$=INKEY$: IF Z$="" TH
EN  GO TO 1260
1270 IF Z$="N" THEN  CLS : STOP
1280 PRINT AT 12,11;"          "
1285 PRINT AT 20,6;"                "
1290 GO SUB 900
1300 PRINT AT 12,3;" 0"
1310 GO TO 200
1400 FOR L=1 TO 8
1410 READ A,B
1420 BEEP A,B
1430 NEXT L
1440 RESTORE 1460
1450 RETURN
1460 DATA .3,7,.1,2,.1,1,.1,2,.3,3,.6,2,
.3,6,.3,7
```

Chapter 9

Tenpin



Description

Tenpin bowling is a game for two players that will be familiar to many readers. The object of the game is to bowl your ball in such a way that you knock over ten skittles at the other end of a long alley. If you knock the ten skittles over in one bowl, you have obtained a 'strike'. This is beneficial to your score because it doubles the points obtained in your next two bowls.

If you are not successful on your first attempt you are allowed a second chance to knock over the remaining skittles. If you remove them all at this stage, this is known as a 'spare'. This also improves your score by doubling the points obtained in your next bowl.

There are ten rounds and you each take turns at bowling. If you should obtain a strike or a spare on your last bowl then you are awarded the appropriate number of extra bowls.

Hints on Entry

Tenpin should prove very easy to enter with the only lines needing special attention being 2000–2099. This is where the ball and the skittles are defined, so numerical errors in the data lines would cause the wrong shapes to be displayed.

When the program is entered then save it on tape or microdrive in the usual way by typing:

```
SAVE "TENPIN" LINE 1
```

or

```
SAVE *"M";1;"TENPIN" LINE 1
```

Techniques

One of the major problems encountered when designing long programs on a Spectrum computer with Interface 1 and a microdrive is the amount of typing required to save the current program onto a cartridge. For example:

```
ERASE "M";1;"PROG"
SAVE *"M";1;"PROG"
VERIFY *"M";1;"PROG"
```

Because of the number of commands, one is often tempted to save the program very infrequently and if machine code routines are used or your Spectrum is prone to "glitch" then the failure to make frequent copies can lead to a lot of unnecessary work.

A simple solution for this problem is to start any program by typing the following line:

```
9999 ERASE "M";1;"PROG":SAVE* "M";1;"PROG":
     VERIFY *"M";1;"PROG"
```

If this is done, then a current copy of the program can be saved onto a cartridge by simply typing GOTO 9999.

Note. Due to the ERASE statement at the beginning of the line, this method can only be used when an original copy has been made.

Instructions

When the program is executed, the bowling alley will be displayed across the centre of the screen with the skittles on the right-hand side. Above and below the alley are the score cards of the two players showing the frame, the number of strikes and spares scored, and the total number of points obtained.

When it is your turn to bowl, the computer will flash your number and ask for a position in the range 1–9. This determines where you start your bowl from, with 1 being the extreme left of the alley and 9 the extreme right. You select your position by pressing the appropriate key and then you are asked for the speed at which you wish to bowl. In this case, 1 is the slowest and 5 the fastest. You are then asked which way you would like to bias the ball—either left or right—and when you have selected either L or R, your ball is automatically bowled.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED BY LOWER CASE LETTERS.

```
  1 POKE 23658,8
  5 DIM B(10)
 10 BORDER 1
 20 GO SUB 1000
 24 LET SC1=0: LET SC2=0: LET ST1=0: LE
T ST2=0: LET SP1=0: LET SP2=0
 25 LET U=0: LET V=0
 30 GO SUB 2000
 40 FOR T=1 TO 10
 50 FOR P=1 TO 2
 51 LET LAST=0
 52 IF P=1 THEN  PRINT AT 17,2;"2": PRI
NT AT 3,1;: FLASH 1: PRINT AT 3,2;"1": F
LASH 0
```

```
  53 IF P=2 THEN  PRINT AT 3,2;"1": PRIN
T AT 17,1;: FLASH 1: PRINT AT 17,2;"2":
FLASH 0
  54 BEEP .1,0: BEEP .1,4: BEEP .1,7: BE
EP 1,12
  55 GO SUB 2100
  60 FOR K=1 TO 10: LET B(K)=0: NEXT K
  65 LET SC=0: LET STR=0: LET SPA=0
  66 GO SUB 1100
  70 GO SUB 1200
  80 GO SUB 1300
  90 GO SUB 1400
 100 LET GO=1: GO SUB 1500
 120 IF STR=1 THEN  GO SUB 1700: GO SUB
2200: GO TO 160
 130 GO SUB 1200
 140 GO SUB 1300
 145 GO SUB 1400
 150 LET GO=2: GO SUB 1500
 155 IF SPA=1 THEN  GO SUB 1800: GO SUB
2200: GO TO 160
 200 GO SUB 1600
 210 PAUSE 50
 220 NEXT P
 230 NEXT T
 235 LET P=1
 236 FOR Q=1 TO 10: LET B(Q)=0: NEXT Q
 238 IF U=0 THEN  GO TO 300
 240 INK 0: PAPER 6: PRINT AT 3,6;"EXTRA
"
 241 IF U=1 THEN  LET UU=1: GO TO 243
 242 LET UU=2
 243 GO SUB 1100: INK 0: PAPER 6
 244 PRINT AT 17,2;"2": PRINT AT 3,1;: F
LASH 1: PRINT AT 3,2;"1": FLASH 0
 245 BEEP .1,0: BEEP .1,4: BEEP .1,7: BE
EP 1,12
 247 LET SPA=0: LET STR=0
 248 FOR W=1 TO UU
```

```
 249 LET SC=0
 250 GO SUB 1200: GO SUB 1300: GO SUB 14
00
 264 FOR Q=1 TO 10: IF B(Q)=1 THEN  LET
SC=SC+1
 265 NEXT Q
 266 IF UU=2 AND W=1 AND SC=10 THEN  GO
SUB 1700: GO SUB 1100: FOR Q=1 TO 10: LE
T B(Q)=0: NEXT Q: LET SC1=SC1+10: LET ST
R=1: GO TO 290
 267 IF W=1 AND SC=10 THEN  GO SUB 1700:
 GO SUB 1100: GO TO 290
 268 IF STR=1 AND W=2 AND SC=10 THEN  GO
 SUB 1700: GO TO 290
 269 IF STR<>1 AND W=2 AND SC=10 THEN  G
O SUB 1800: GO TO 290
 290 NEXT W
 292 LET SC1=SC1+SC
 294 GO SUB 1600
 300 LET P=2
 305 FOR Q=1 TO 10: LET B(Q)=0: NEXT Q
 310 IF U=0 THEN  GO TO 400
 312 INK 0: PAPER 6: PRINT AT 17,6;"EXTR
A"
 314 IF U=1 THEN  LET UU=1: GO TO 320
 315 LET UU=2
 320 GO SUB 1100: INK 0: PAPER 6
 325 PRINT AT 3,2;"1": PRINT AT 17,1;: F
LASH 1: PRINT AT 17,2;"2": FLASH 0
 330 BEEP .1,0: BEEP .1,4: BEEP .1,7: BE
EP 1,12
 334 LET STR=0: LET SPA=0
 335 FOR W=1 TO UU
 340 LET SC=0
 345 GO SUB 1200: GO SUB 1300: GO SUB 14
00
 350 FOR Q=1 TO 10: IF B(Q)=1 THEN  LET
SC=SC+1
```

```
 351 NEXT Q
 355 IF VV=2 AND W=1 AND SC=10 THEN  GO
SUB 1700: GO SUB 1100: FOR Q=1 TO 10: LE
T B(Q)=0: NEXT Q: LET SC2=SC2+10: LET ST
R=1: GO TO 390
 356 IF W=1 AND SC=10 THEN  GO SUB 1700:
 GO SUB 1100: GO TO 390
 360 IF STR=1 AND W=2 AND SC=10 THEN  GO
 SUB 1700: GO TO 390
 365 IF STR<>1 AND W=2 AND SC=10 THEN  G
O SUB 1800: GO TO 390
 390 NEXT W
 392 LET SC2=SC2+SC: GO SUB 1600
 400 INK 0: PAPER 6: PRINT AT 3,2;"1": P
RINT AT 17,2;"2"
 401 GO SUB 1100
 402 PAPER 7: INK 0
 405 IF SC1>SC2 THEN  PRINT AT 9,8;"PLAY
ER 1 WINS"
 410 IF SC1<SC2 THEN  PRINT AT 9,8;"PLAY
ER 2 WINS"
 415 IF SC1=SC2 THEN  PRINT AT 9,8;"MATC
H DRAWN"
 420 PAUSE 100
 425 PAPER 5: PRINT AT 20,7;"ANOTHER GAM
E (Y/N)"
 430 LET A$=INKEY$: IF A$="" THEN  GO TO
 430
 435 IF A$="Y" THEN  RUN
 440 CLS : STOP
1000 FOR I=0 TO 21
1005 IF I>18 THEN  PAPER 5: GO TO 1030
1010 IF I<5 OR I>13 THEN  PAPER 6: GO TO
 1030
1020 PAPER 7
1030 PRINT AT I,0;"
                  "
1040 NEXT I
1041 PAPER 7: FOR Q=1 TO 9: PRINT AT 4+Q
```

```
,0;Q;" ";CHR$ 138: NEXT Q
1045 PAPER 6: INK 0
1050 FOR J=0 TO 14 STEP 14
1055 PRINT AT 1+J,1;"PLY"
1058 PRINT AT 1+J,6;"FRAME"
1060 PRINT AT 1+J,12;"STRIKE"
1065 PRINT AT 1+J,19;"SPARE"
1070 PRINT AT 1+J,25;"SCORE"
1075 PRINT AT 3+J,2;1+INT (J/14)
1078 PRINT AT 3+J,8;"0"
1080 PRINT AT 3+J,15;"0"
1085 PRINT AT 3+J,21;"0"
1090 PRINT AT 3+J,28;"0"
1095 NEXT J
1097 RETURN
1100 INK 3: PAPER 7: PRINT AT 9,27;CHR$
144
1105 PRINT AT 8,28;CHR$ 144
1110 PRINT AT 10,28;CHR$ 144
1115 PRINT AT 7,29;CHR$ 144
1120 PRINT AT 9,29;CHR$ 144
1125 PRINT AT 11,29;CHR$ 144
1130 PRINT AT 6,30;CHR$ 144
1135 PRINT AT 8,30;CHR$ 144
1140 PRINT AT 10,30;CHR$ 144
1145 PRINT AT 12,30;CHR$ 144
1150 RETURN
1200 INK 0: PAPER 5: PRINT AT 20,7;"POSI
TION ( 1-9 )"
1205 PAUSE 0: LET P$=INKEY$: IF P$="" TH
EN  GO TO 1205
1210 IF CODE P$<49 OR CODE P$>57 THEN  G
O TO 1205
1215 LET POS=VAL P$
1220 PRINT AT 20,7;"  SPEED ( 1-5 )  "
1225 PAUSE 0: LET S$=INKEY$: IF S$="" TH
EN  GO TO 1225
1230 IF CODE S$<49 OR CODE S$>53 THEN  G
O TO 1225
```

```
1235 LET S=VAL S$
1240 PRINT AT 20,7;"  BIAS ( L OR R )"
1245 PAUSE 0: LET B$=INKEY$: IF B$="" TH
EN  GO TO 1245
1250 IF B$<>"L" AND B$<>"R" THEN  GO TO
1245
1255 PRINT AT 20,7;"                    "
1260 RETURN
1300 INK 4: PAPER 7: LET X=4+POS: LET Z=
0
1305 FOR Y=3 TO 31
1310 IF INT (30*RND)<S+23 THEN  GO TO 13
40
1315 IF B$="L" THEN  LET X=X-1
1320 IF B$="R" THEN  LET X=X+1
1325 IF X<5 THEN  LET X=5
1330 IF X>13 THEN  LET X=13
1340 IF Y<>3 THEN  PRINT AT X1,Y1;" "
1345 PRINT AT X,Y;CHR$ 145
1350 LET X1=X: LET Y1=Y
1352 LET PA=5-S: IF PA=0 THEN  GO TO 135
5
1353 PAUSE PA
1355 IF Y<27 OR Z<>0 THEN  GO TO 1385
1358 IF Y=27 AND X=9 AND B(1)=0 THEN  LE
T Z=1: GO TO 1385
1360 IF Y=28 AND X=8 AND B(2)=0 THEN  LE
T Z=2: GO TO 1385
1362 IF Y=28 AND X=10 AND B(3)=0 THEN  L
ET Z=3: GO TO 1385
1364 IF Y=29 AND X=7 AND B(4)=0 THEN  LE
T Z=4: GO TO 1385
1366 IF Y=29 AND X=9 AND B(5)=0 THEN  LE
T Z=5: GO TO 1385
1368 IF Y=29 AND X=11 AND B(6)=0 THEN  L
ET Z=6: GO TO 1385
1370 IF Y=30 AND X=6 AND B(7)=0 THEN  LE
T Z=7: GO TO 1385
1372 IF Y=30 AND X=8 AND B(8)=0 THEN  LE
T Z=8: GO TO 1385
```

```
1374 IF Y=30 AND X=10 AND B(9)=0 THEN  L
ET Z=9: GO TO 1385
1376 IF Y=30 AND X=12 AND B(10)=0 THEN
LET Z=10: GO TO 1385
1385 NEXT Y
1386 PRINT AT X,Y-1;" "
1387 IF Z=0 THEN  BEEP .1,7: BEEP .1,7:
BEEP .1,7: BEEP .5,3: BEEP .1,5: BEEP .1
,5: BEEP .1,5: BEEP .5,2: RETURN
1389 LET B(Z)=1
1390 IF Z=1 AND RND<.5 THEN  FOR Q=1 TO
10: LET B(Q)=1: NEXT Q: GO TO 1396
1392 GO SUB 2300
1393 FOR K=Y TO 10
1394 IF RND<.7 THEN  LET B(K)=1
1396 NEXT K
1398 RETURN
1400 INK 3: PAPER 7
1403 FOR J=0 TO 1
1405 IF B(1)=J THEN  PRINT AT 9,27;CHR$
(144-16*J)
1410 IF B(2)=J THEN  PRINT AT 8,28;CHR$
(144-16*J)
1415 IF B(3)=J THEN  PRINT AT 10,28;CHR$
 (144-16*J)
1420 IF B(4)=J THEN  PRINT AT 7,29;CHR$
(144-16*J)
1425 IF B(5)=J THEN  PRINT AT 9,29;CHR$
(144-16*J)
1430 IF B(6)=J THEN  PRINT AT 11,29;CHR$
 (144-16*J)
1435 IF B(7)=J THEN  PRINT AT 6,30;CHR$
(144-16*J)
1440 IF B(8)=J THEN  PRINT AT 8,30;CHR$
(144-16*J)
1445 IF B(9)=J THEN  PRINT AT 10,30;CHR$
 (144-16*J)
1450 IF B(10)=J THEN  PRINT AT 12,30;CHR
$ (144-16*J)
```

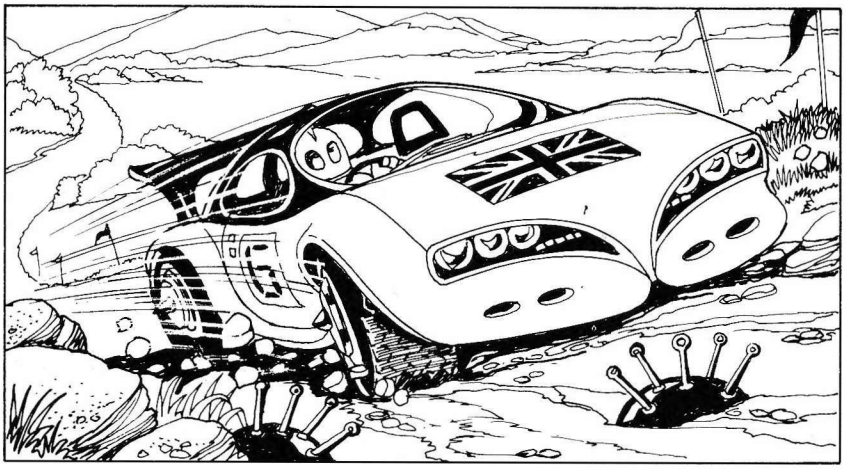
```
1455 NEXT J
1460 RETURN
1500 LET SC=0: FOR K=1 TO 10
1505 IF B(K)=1 THEN  LET SC=SC+1
1510 NEXT K
1515 IF LAST<>0 THEN  LET SC=SC-LAST
1520 IF P=2 THEN  GO TO 1545
1522 IF U=0 THEN  LET SC1=SC1+SC: GO TO
1570
1525 IF U>1 THEN  LET SC1=SC1+U*SC: LET
U=1: GO TO 1570
1530 LET SC1=SC1+2*SC: LET U=0: GO TO 15
70
1545 IF V=0 THEN  LET SC2=SC2+SC: GO TO
1570
1550 IF V>1 THEN  LET SC2=SC2+V*SC: LET
V=1: GO TO 1570
1555 LET SC2=SC2+2*SC: LET V=0: GO TO 15
70
1570 IF SC=10 AND GO=1 THEN  LET STR=1
1580 IF LAST+SC=10 AND GO=2 THEN  LET SP
A=1
1585 LET LAST=SC
1590 RETURN
1600 PAPER 6: INK 0: IF P=2 THEN  GO TO
1650
1605 IF SC1<10 THEN  PRINT AT 3,28;SC1:
GO TO 1610
1606 IF SC1<100 THEN  PRINT AT 3,27;SC1:
 GO TO 1610
1607 IF SC1<1000 THEN  PRINT AT 3,26;SC1
: GO TO 1610
1649 RETURN
1650 IF SC2<10 THEN  PRINT AT 17,28;SC2:
 GO TO 1660
1655 IF SC2<100 THEN  PRINT AT 17,27;SC2
: GO TO 1660
1657 IF SC2<1000 THEN  PRINT AT 17,26;SC
2: GO TO 1660
```

```
1699 RETURN
1700 FLASH 1: INK 0: PAPER 7: PRINT AT 9
,26;"STRIKE"
1701 FOR Q=0 TO 50: BEEP .01,Q: NEXT Q:
FLASH 0: PRINT AT 9,26;"      "
1702 PAPER 6: IF P<>1 THEN  GO TO 1750
1705 LET ST1=ST1+1
1710 IF ST1<10 THEN  PRINT AT 3,15;ST1:
RETURN
1720 PRINT AT 3,14;ST1: RETURN
1750 LET ST2=ST2+1
1751 IF ST2<10 THEN  PRINT AT 17,15;ST2:
 RETURN
1760 PRINT AT 17,14;ST2: RETURN
1800 FLASH 1: INK 0: PAPER 7: PRINT AT 9
,27;"SPARE"
1801 FOR Q=50 TO 0 STEP -1: BEEP .01,Q:
NEXT Q: FLASH 0: PRINT AT 9,27;"     "
1802 PAPER 6: IF P<>1 THEN  GO TO 1850
1805 LET SP1=SP1+1
1810 IF SP1<10 THEN  PRINT AT 3,21;SP1:
RETURN
1820 PRINT AT 3,20;SP1: RETURN
1850 LET SP2=SP2+1
1851 IF SP2<10 THEN  PRINT AT 17,21;SP2:
 RETURN
1860 PRINT AT 17,20;SP2: RETURN
2000 FOR N=0 TO 7
2010 READ R
2020 POKE USR "a"+N,R
2030 NEXT N
2040 FOR N=0 TO 7
2050 READ R
2060 POKE USR "b"+N,R
2070 NEXT N
2080 RETURN
2098 DATA 16,16,56,56,56,124,124,56
2099 DATA 0,0,0,62,127,127,127,62
2100 IF P=2 THEN  GO TO 2150
```

```
2110 IF T<10 THEN  PRINT AT 3,8;T: RETUR
N
2120 PRINT AT 3,7;T: RETURN
2150 IF T<10 THEN  PRINT AT 17,8;T: RETU
RN
2160 PRINT AT 17,7;T: RETURN
2200 IF P=1 AND STR=1 THEN  LET U=U+2
2210 IF P=1 AND SPA=1 THEN  LET U=U+1
2220 IF P=2 AND STR=1 THEN  LET V=V+2
2230 IF P=2 AND SPA=1 THEN  LET V=V+1
2240 RETURN
2300 IF Z=1 THEN  LET Y=1
2310 IF Z=2 OR Z=3 THEN  LET Y=2
2320 IF Z=4 OR Z=5 OR Z=6 THEN  LET Y=4
2340 IF Z=7 OR Z=8 OR Z=9 OR Z=10 THEN
LET Y=7
2350 RETURN
```

Chapter 10

The Madcap Rally



Description

The year is 2032, and the main event at the London Olympics is the deadly Madcap Rally. Each competitor is driving a Mark 3 Sinclair Electric Buggy, and his task is to go round the circuit collecting all the markers shown in green and blue, but avoiding the anti-buggy mines indicated in red.

Each country entering the event has only one competitor, and five buggies equipped with a constant-speed motor, high-sensitivity steering but, sadly, no brakes! Four days into the event, it finally becomes your round. The Union Jack is raised, the anthem is played and you're off in search of gold.

Hints on Entry

As can be seen in the listing below, the game consists of two programs. These must be entered and then saved in the correct order on the tape or microdrive if the program is to operate in the correct manner.

1) The first program, Rally, is typed into the computer, paying particular attention to the user-defined graphics. This program is then saved on tape or microdrive using the command:

   SAVE "RALLY" LINE 1

or

   SAVE *"M";1;"RALLY" LINE 1

2) The second short program, used to define the graphics characters, is now entered, taking care with the lines of data. When the program has been entered, typing RUN will cause the graphics characters to be designed with the computer storing the information in RAM, starting at the location held in the system variable UDG. When the generation process is complete, the information relating to these characters will be saved on tape for future use by the main program.

Note. Although the characters are saved on tape, the program is not, and as this program will be required for the next game it should also be saved by typing:

SAVE "GRAPHICS"

or

SAVE *"M";1;"GRAPHICS"

Techniques

One problem often experienced with very long games, requiring a large number of user-defined graphics, is the lack of available memory, as the amount of space required to define a set of graphics characters is often considerable. In such a case, or when it is necessary to use the same characters for several games, it is useful to design the graphics using a separate program and then save them on tape using the SAVE CODE function.

The SAVE CODE function, as defined in the Spectrum manual, can be used to save blocks of memory on tape or microdrive. The syntax of the instruction is:

SAVE"NAME" CODE START,LENGTH

where NAME is the name to be recorded on tape; START is the first memory location of the block; and LENGTH is the number of bytes to be saved. In the particular case of saving the user-defined graphics, the name is a matter of personal choice, start has to be calculated by using the system variable UDG stored at locations 23675 and 23676, and the length is calculated by multiplying the number of characters by eight.

Example

If we have created ten characters then these could be saved onto tape using:

SAVE "CHARS" CODE (23675 + (256*23676)),80

Instructions

The game is loaded from tape or microdrive by typing:

LOAD ""

or

LOAD *"M";1;"RALLY"

When loaded the game will run automatically. There will be a short delay while the graphics characters are loaded into RAM after which a series of instructions will be displayed on the screen. The game then commences and after inputting the required level of difficulty (10–300) you must use the cursor keys to steer your buggy around the track, collecting first the green and then the blue markers.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED BY LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
   3 LOAD ""CODE
   4 LET HS=0
   5 BORDER 0: PAPER 0: CLS
  10 PRINT AT 10,8; INK 7; PAPER 2; BRIG
HT 1; FLASH 1;"THE MADCAP RALLY"
  23 FOR N=0 TO 7: BORDER N: FOR F=-30 T
O 30 STEP 5: BEEP N/100,F: NEXT F: NEXT
N
  24 BORDER 0
  25 INPUT "PRESS ENTER TO CONTINUE"; LI
NE Y$
  30 GO SUB 8000
  40 GO SUB 7000
  50 CLS
  52 IF SK>30 THEN  PRINT AT 10,10; FLAS
H 1;"PLEASE WAIT"
  55 LET LIVES=5
  60 LET PLX=10: LET PLY=10
  70 LET C$="e"
 100 DIM X(SK): DIM Y(SK)
 105 DIM U(SK): DIM V(SK)
 107 LET K=SK
 108 LET B=INT (RND*20)+1: LET C=INT (RN
D*30)+1
 110 FOR N=1 TO SK
 120 LET X(N)=INT (RND*20)+1
 125 LET U(N)=INT (RND*20)+1
 130 LET Y(N)=INT (RND*29)+1
 135 LET V(N)=INT (RND*30)+1
 140 NEXT N
 145 CLS
 200 FOR N=1 TO SK
 210 PRINT AT X(N),Y(N); INK 4;"b"
 220 PRINT AT U(N),V(N); INK 2;"b"
 230 NEXT N
 235 PRINT AT B,C; INK 1;"b"
 237 PRINT AT PLX,PLY; INK 5;C$: PAUSE 0
 240 PRINT AT PLX,PLY; INK 5;C$
 245 LET OPLY=PLY: LET OPLX=PLX
```

```
 247 BEEP .001,RND*50
 250 IF INKEY$="7" THEN  LET C$="c"
 260 IF INKEY$="6" THEN  LET C$="i"
 270 IF INKEY$="5" THEN  LET C$="g"
 280 IF INKEY$="8" THEN  LET C$="e"
 285 IF C$="e" THEN  LET PLY=PLY+(1 AND
PLY<31)
 286 IF C$="g" THEN  LET PLY=PLY-(1 AND
PLY>0)
 287 IF C$="c" THEN  LET PLX=PLX-(1 AND
PLX>0)
 288 IF C$="i" THEN  LET PLX=PLX+(1 AND
PLX<21)
 289 IF ATTR (PLX,PLY)=4 THEN  LET K=K-1
 290 IF ATTR (PLX,PLY)=2 THEN  GO SUB 10
00
 291 IF ATTR (PLX,PLY)=1 THEN  GO TO 200
0
 292 IF OPLX<>PLX OR OPLY<>PLY THEN  PRI
NT AT OPLX,OPLY;" "
 300 GO TO 240
1000 PRINT AT OPLX,OPLY; PAPER 2; INK 7;
 BRIGHT 1; FLASH 1;C$
1005 FOR N=0 TO 8 STEP .1: BORDER INT N:
 BEEP .01,N: NEXT N
1006 BORDER 0
1010 LET LIVES=LIVES-1
1020 IF LIVES=0 THEN  GO TO 2000
1025 IF INKEY$<>"" THEN  GO TO 1025
1027 IF INKEY$="" THEN  GO TO 1027
1030 RETURN
2000 PRINT AT PLX,PLY-INT (PLY/5); FLASH
 1; INK 6;"GAME OVER"
2005 FOR N=1 TO 100: NEXT N
2007 BORDER 5: PAPER 5: INK 0: CLS
2010 LET SC=INT 50*((SK*5)-(K*5)+(LIVES*
10))
2020 PRINT AT 0,0;"YOUR SCORE IS "; FLAS
H 1; PAPER 2; INK 7;SC
```

```
2025 IF SC>HS THEN  PRINT '' FLASH 1; PA
PER 2; INK 7;"YOU HAVE GOT THE HIGH SCOR
E !!!!": LET HS=SC
2027 PRINT AT 21,0;"THE HIGH SCORE IS ";
 FLASH 1; PAPER 2; INK 7;HS
2030 INPUT "ANOTHER GAME ?";Y$
2040 IF Y$="Y" THEN  PAPER 0: BORDER 0:
CLS : INK 7: GO TO 40
3000 STOP
7000 INPUT "DIFFICULTY LEVEL ?(10 TO 300
)";SK
7010 IF SK<10 THEN  GO TO 7000
7020 RETURN
8000 INK 7: CLS : PRINT AT 0,0; PAPER 2;
 INK 7; FLASH 1;"THE MADCAP RALLY"
8010 PRINT '''  IN THIS GAME YOU MUST ST
EER  YOUR BLUE CAR AROUND THE SCREEN,US
ING THE ARROW KEYS."
8020 PRINT '"  YOU MUST RUN OVER THE GRE
EN  OBSTACLES AND TRY TO AVOID THE  RED
 ONES. IF YOU HIT A RED ONE  YOU WILL LO
SE A LIFE."
8030 PRINT '"  ONCE YOU HAVE RUN OVER AL
L THEGREEN THINGS, YOU MUST GO OVER  THE
 BLUE ONE TO END THE GAME."
8040 PRINT "IF YOU HIT THE BLUE ONE BY
    ACCIDENT, IT WILL END THE GAME."
8050 PRINT '"  YOU HAVE 5 LIVES  AT THE
     START. YOUR CAR WILL CARRY ON  MOV
ING IN THE DIRECTION CHOSEN."
8999 RETURN
```

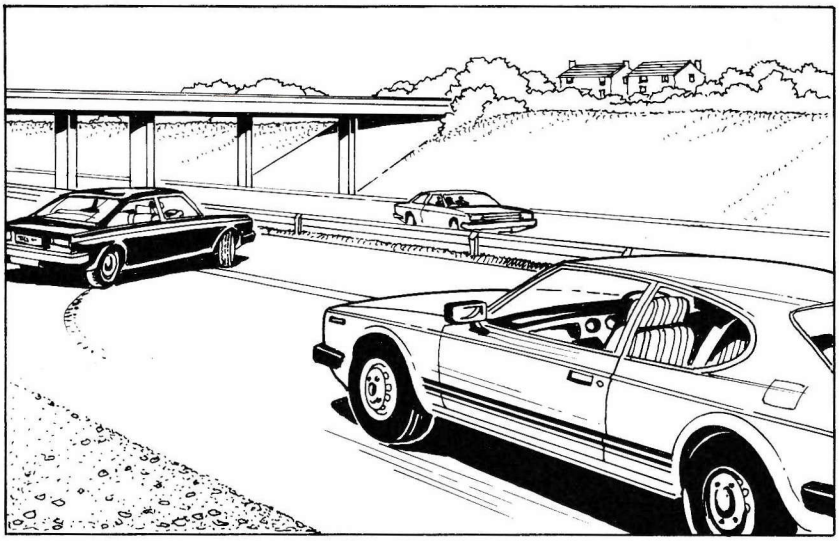
Graphics

```
  10 REM CHARACTERS FOR CAR GAMES
  20 FOR N=1 TO 5: READ A$: LET as=USR A
$: FOR X=0 TO 7: READ E: POKE AS+X,E: NE
XT X: NEXT N
```

```
  30 DATA "B",24,60,126,255,255,126,60,2
4
  40 DATA "C",16,186,186,40,56,214,238,2
14
  50 DATA "E",0,230,224,94,183,94,224,23
0
  60 DATA "G",0,103,7,122,237,122,7,103
  70 DATA "I",107,119,107,28,20,93,93,81
  90 SAVE "CHARS"CODE (PEEK 23675+(256*P
EEK 23676)),72
```

Chapter 11

Road Racer



Description

Having just won the gold medal in the Madcap Rally you must now drive your Sinclair Electric Buggy back along the M1 to your home in the suburbs of Birmingham. As always the motorway is overcrowded, and half the road users are either asleep or totally mad; so can you get home alive without wrecking your very expensive car? Only time will tell.

Hints on Entry

Inspection of the listing given below shows that the game is in two parts: the main program which should be entered first and

saved on tape, followed by a short program used to define and save on tape the data relating to the user-defined graphics.

If the instructions for the Madcap Rally have been adhered to correctly, then you will already have a copy of the second program on tape or microdrive. Therefore, to save time and the trouble of re-entering, load this program and type RUN. The redefined character information can then be saved onto the Road Racer tape for future use.

Techniques

One of the most powerful commands available to the games programmer in Sinclair BASIC is the ATTR function, which returns information concerning the character in a specified position on the screen.

If we type in a statement such as PRINT ATTR(10,10), the computer will return a number in the range 0-255, and this number can be decoded to give information relating to the flashing status, bright status, paper colour and ink colour. To extract information from the number, we need first to understand how the information is stored inside the computer and secondly how the data is arranged within the word.

In a computer, all numbers are represented in binary form by a byte (a group of eight bits). In the case of the attribute bytes, the information is held according to the following format:

	FLASH	BRIGHT	PAPER	PAPER	PAPER	INK	INK	INK
BIT	7	6	5	4	3	2	1	0

Examples

IF ATTR(X,Y) = 136 = 10001000
then

INK	0	(BLACK)
PAPER	1	(BLUE)
BRIGHT	0	(NORMAL)
FLASH	1	(FLASHING)

IF ATTR(X,Y) = 219 = 11011011
then

INK	3	(MAGENTA)
PAPER	3	(MAGENTA)
BRIGHT	1	(BRIGHT)
FLASH	1	(FLASHING)

From these examples we see that the information is coded in a complex form. Obtaining information concerning the individual paper colour, ink colour or status can therefore involve several lines of program.

Example

To find the paper colour at character position (10,10) we would require:

```
10 LET A = ATTR(10,10)
20 IF A > 63 THEN LET A + A – 64: GOTO 20
30 LET P = INT(A/8)
40 REM THE VALUE OF 'P' IS NOW THE PAPER COLOUR
   AT (10,10)
```

Instructions

The game is loaded from tape or microdrive by typing:

```
LOAD ""
```

or

```
LOAD* "M";1;"ROAD"
```

When loaded the game will run automatically. The tape machine must be left on so that the graphics characters, saved in code form, can also be loaded into the computer. When this has been completed the instructions will appear on the screen, and after selecting the level of difficulty the game will commence with the car being controlled by the two cursor keys: KEY 5, move left; key 8, move right.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED BY LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
   1 LOAD ""CODE
   2 POKE 23658,8
  10 CLS : PRINT TAB 10; FLASH 1; PAPER
2; INK 7; BRIGHT 1;"ROAD RACER"
  15 FOR N=1 TO 5: FOR F=0 TO 7: BORDER
F: BEEP .05,F+N: BEEP .05,F-N: NEXT F: N
EXT N
  16 BORDER 0: BRIGHT 1: PRINT #0;"PRESS
 ENTER TO CONTINUE": PAUSE 0
  17 PAPER 0: BORDER 0: CLS : INK 7
  20 GO SUB 9000
  30 GO SUB 6000
  40 PRINT AT 10,X; INK 6;"i"
  44 FOR N=1 TO 6: NEXT N
  45 LET OX=X
  46 LET S=S+1
  50 LET CX=INT (RND*(SK-1))+11
  60 PRINT AT 21,10;"b";AT 21,SK+10;"b"
  70 PRINT AT 21,CX; INK 5;"c";AT 21,31;
" "
  77 LET L=USR 3582
  78 PRINT AT 9,OX;" "
  80 IF ATTR (11,X)=69 THEN  GO SUB 1000
  90 LET X=X+(INKEY$="8" AND X<SK+9)-(IN
KEY$="5" AND X>11)
 100 GO TO 40
1000 PRINT AT 10,X; FLASH 1; INK 2;"o"
1005 FOR N=0 TO 7: BORDER N: BEEP .05,N+
RND*30: BEEP .25,N: NEXT N
1007 BORDER 0
```

```
1010 FOR N=1 TO 50: NEXT N: PRINT AT 10,
X;" "
1012 LET LI=LI-1
1015 IF LI=0 THEN  GO TO 2000
1020 RETURN
2000 PRINT AT 10,11; FLASH 1;"GAME OVER"
: FOR N=1 TO 500: NEXT N
2005 FOR M=1 TO 10: FOR N=0 TO 7: BORDER
 N: BEEP .05,N+(RND*50)-(RND*50): NEXT N
: NEXT M
2007 BORDER 0
2010 CLS : PRINT "YOU SCORED ";: FLASH 1;
S
2020 IF S>HS THEN  LET HS=S: GO SUB 3000
2030 PRINT "THE HIGH SCORE IS "; FLASH 1
;HS: PRINT ''"AND IT WAS GOT BY "; FLASH
 1;N$
2040 INPUT "ANOTHER GAME ?(Y/N)";X$
2050 IF X$="Y" THEN  GO TO 30
2060 STOP
3000 PRINT '''''"YOU HAVE GOT THE HIGH S
CORE!": INPUT "INPUT YOUR NAME:";N$
3010 CLS : RETURN
6000 LET X=14
6001 LET LI=5
6005 RESTORE 6100: FOR N=65000 TO 65000+
17: READ A: POKE N,A: NEXT N
6007 LET S=0
6010 LET SC=0: LET HS=0: LET N$="ZX-SPEC
TRUM"
6020 PRINT AT 0,0;"SELECT A SKILL LEVEL
(3 TO 10)"
6030 INPUT SK: IF SK>10 OR SK<3 THEN  GO
 TO 6030
6100 DATA 62,0,17,0,88,6,24,197,6,32,18,
19,16,252,193,16,246,201
6110 RETURN
9000 CLS : PRINT  PAPER 2,"ROAD RACER"
9010 PRINT '" IN THIS GAME YOU ARE DRIVI
```

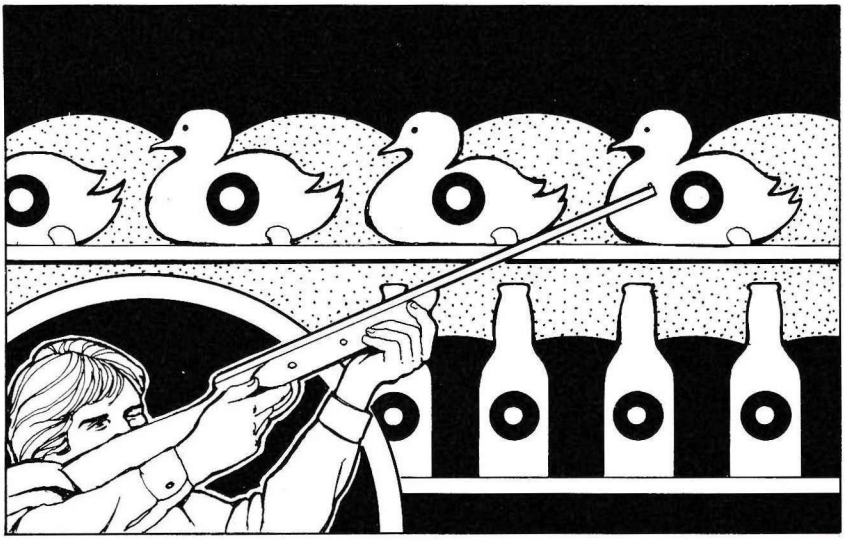
```
NG A BUGGY THE WRONG WAY DOWN A MOTORWAY
. THE ONLY PROBLEM IS:ALL THE OTHER CARS
ARE DRIVING THE OTHERWAY! YOU MUST TRY T
O AVOID       CONTACT FOR AS LONG AS POSS
IBLE."
9020 PRINT '''' USE '5' TO MOVE YOUR CAR
LEFT"''" USE '8' TO MOVE YOUR CAR RIGHT"
9030 PRINT #0;"PRESS ENTER TO START": PA
USE 0
9040 CLS : RETURN
9999 RETURN
```

Graphics

```
  10 REM CHARACTERS FOR CAR GAMES
  20 FOR N=1 TO 5: READ A$: LET as=USR A
$: FOR X=0 TO 7: READ E: POKE AS+X,E: NE
XT X: NEXT N
  30 DATA "B",24,60,126,255,255,126,60,2
4
  40 DATA "C",16,186,186,40,56,214,238,2
14
  50 DATA "E",0,230,224,94,183,94,224,23
0
  60 DATA "G",0,103,7,122,237,122,7,103
  70 DATA "I",107,119,107,28,20,93,93,81
  90 SAVE "CHARS"CODE (PEEK 23675+(256*P
EEK 23676)),72
```

Chapter 12

Target Practice



Description

So you think you're a good shot, eh? Well now is your chance to find out in the Sinclair shooting gallery. The object of the game is to score as many points as possible by hitting the ducks, clay pigeons, bottles and cans. The game will require quick reflexes, a good eye and a fast brain, because if you can score exactly 1000 points then you obtain some bonus shots. Good luck and good shooting.

Hints on Entry

The program is not too long and contains only a few user-defined graphics which are represented in the usual manner, by using lower case letters.

When the program has been entered it should be saved on tape or microdrive by typing:

```
SAVE "TARGET" LINE 1
```

or

```
SAVE *"M";1;"TARGET" LINE 1
```

Techniques

One of the most useful applications of the IN statement is checking for combinations of keys being pressed on the keyboard. It is possible to check if one key is pressed by using the INKEY$ function, but this has the major drawback that it will not test for combinations of keys. This leads to difficulties if we are trying to use four keys to represent eight possible directions of movement.

Consider the problem of using the four keys Q,W,E,R to control a pixel on the screen which is permitted to move in any one of eight directions. The four keys are serviced by the command IN 64510 (see Table 1) with the four least significant bits being used to indicate whether or not Q,W,E or R is pressed. For example,

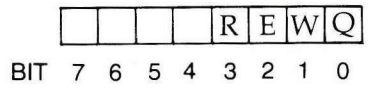


Table 1

Command	Keys
IN 65278	CAPS SHIFT to V
IN 65022	A to G
IN 64510	Q to T
IN 63486	1 to 5
IN 61438	0 to 6
IN 57342	P to Y
IN 49150	ENTER to H
IN 32766	SPACE to B

Having obtained our byte of information, we must now extract information from the relevant bits. This can be achieved by using the following routine:

```
9000 REM CHECK KEYS
9010 LET Q=0: LET W=0: LET R=0: LET E=0
9020 LET A=IN 64510
9025 PRINT A
9030 LET A=255-A
9040 IF A>15 THEN  LET A=A-16: GO TO 904
0
9050 IF A>7 THEN  LET R=1: LET A=A-8
9060 IF A>3 THEN  LET E=1: LET A=A-4
9070 IF A>1 THEN  LET W=1: LET A=A-2
9080 IF A=1 THEN  LET Q=1
9090 RETURN
```

When the computer returns from the subroutine at line 9090, the variables Q,W,E and R hold the information determining whether the key has been pressed and we can then use Boolean expressions such as IF Q THEN to act according to the information.

Instruction

The game is loaded from tape or microdrive by typing:

```
LOAD ""
```

or

```
LOAD *"M";1;"TARGET"
```

When the loading process is completed, the game will run automatically and a set of instructions will be displayed on the screen.

The object of the game is to use the cursor keys to line up your sight with the various targets in the shooting gallery and then use the B key to fire. A bonus can be obtained if the total number of points scored at the end of the round is exactly 1000.

Listing

IMPORTANT

1) THIS GAME SHOULD BE ENTERED USING CAPS LOCK.

2) ALL GRAPHICS CHARACTERS ARE INDICATED BY LOWER CASE LETTERS.

3) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

```
  1 POKE 23658,8
  5 BORDER 1
  6 LET HS=0
  7 LET N$="ZX-SPECTRUM"
 10 LET INSTR=8000
 20 GO SUB INSTR: REM (LONG OR SHORT IN
STRUCTIONS)
 30 GO SUB 4000: REM VARIABLES
 40 GO SUB 5000: REM PICTURE
100 LET R=INT (RND*4)+1: IF R=1 THEN  G
O SUB 1000
110 IF R=2 THEN  GO SUB 1200
120 IF R=3 THEN  GO SUB 1400
130 IF R=4 THEN  GO SUB 1600
140 GO TO 100
200 OVER 0: INK 0: PLOT  INK 9;PX-20,PY
: DRAW 40,0: PLOT  INK 9;PX,PY-20: DRAW
0,40
205 OVER 0: INK 0: PLOT PX-20,PY: DRAW
40,0: PLOT PX,PY-20: DRAW 0,40
206 GO SUB 5020
207 LET OPX=PX: LET OPY=PY
210 LET PX=PX+(8*(INKEY$="8" AND PX<234
))-(8*(INKEY$="5" AND PX>21))
220 LET PY=PY+(8*(INKEY$="7" AND PY<150
))-(8*(INKEY$="6" AND PY>21))
225 IF OPX<>PX OR OPY<>PY THEN  OVER 1:
PLOT OPX-20,OPY: DRAW 40,0: PLOT OPX,OP
Y-20: DRAW 0,19: PLOT OPX,OPY+1: DRAW 0,
19
```

```
 227 IF INKEY$="B" OR INKEY$="B" THEN  G
O SUB 2000
 228 OVER 0: PAPER 2: INK 7: PRINT AT 21
,12-LEN STR$ SC;SC;AT 21,18;("0" AND AMM
O<10);AT 21,20-LEN STR$ AMMO;AMMO;AT 21,
30-LEN STR$ HS;HS: PAPER 5: INK 0
 230 RETURN
1000 GO SUB 200
1005 LET C=1
1010 LET CX=INT (RND*4)+8: LET CY=INT (R
ND*6)+10
1012 FOR N=1 TO 10
1015 GO SUB 200
1020 OVER 0: PRINT AT CX,CY;"c"
1025 IF HIT=1 THEN  LET HIT=0: GO TO 104
0
1030 NEXT N
1040 PRINT AT CX,CY; OVER 1;"c": OVER 0
1045 LET C=0
1047 LET CX=0: LET CY=0
1050 RETURN
1200 GO SUB 200
1205 LET B=1
1210 LET CX=INT (RND*3)+9: LET CY=INT (R
ND*7)+11
1212 FOR N=1 TO 10
1215 GO SUB 200
1220 OVER 0: PRINT AT CX,CY;"d"
1225 IF HIT=1 THEN  LET HIT=0: GO TO 124
0
1230 NEXT N
1240 PRINT AT CX,CY; OVER 1;"d": OVER 0
1245 LET B=0
1247 LET CX=0: LET CY=0
1250 RETURN
1400 GO SUB 200
1405 LET D=1
1410 LET Y=4: FOR X=6 TO 25
1415 PRINT AT Y,X;("a" AND X/2=INT (X/2)
```

```
);("b" AND X/2<>INT (X/2))
1416 BEEP .01,0
1417 LET CX=Y: LET CY=X
1420 GO SUB 200
1430 IF HIT=1 THEN  LET HIT=0: GO TO 144
0
1432 PRINT AT CX,CY; OVER 1;("a" AND CY/
2=INT (CY/2)); OVER 1;("b" AND CY/2<>INT
 (CY/2))
1435 NEXT X
1437 OVER 1: PRINT AT Y,X;("a" AND X/2=I
NT (X/2));("b" AND X/2<>INT (X/2))
1440 PRINT AT Y,X; OVER 0;" "
1445 LET D=0
1447 LET CX=0: LET CY=0
1450 RETURN
1600 GO SUB 200
1605 LET Q=1
1610 LET Y=3: FOR X=4 TO 23
1615 PRINT AT Y,X;"e"
1616 BEEP .01,5
1617 LET CX=Y: LET CY=X
1620 GO SUB 200
1630 IF HIT=1 THEN  LET HIT=0: GO TO 164
0
1632 PRINT AT CX,CY; OVER 1;"e"
1635 NEXT X
1640 PRINT AT Y,X; OVER 0;" "
1645 LET Q=0
1647 LET CX=0: LET CY=0
1650 RETURN
2000 BORDER 2
2005 BEEP .05,-10
2007 LET AMMO=AMMO-1: IF AMMO=0 THEN  GO
 TO 2200
2010 PLOT PX,PY: DRAW 255-PX,-PY: PLOT P
X-1,PY-1: DRAW -PX+1,-PY+1
2015 BEEP .01,30
2020 IF INT (PX/8)=CY AND INT (22-(PY/8)
```

```
)=CX THEN  GO TO 2040
2030 OVER 1: PLOT PX,PY: DRAW 255-PX,-PY
: PLOT PX-1,PY-1: DRAW -PX+1,-PY+1: OVER
 0: BORDER 1: RETURN
2040 OVER 1: PLOT PX,PY: DRAW 255-PX,-PY
: PLOT PX-1,PY-1: DRAW -PX+1,-PY+1: OVER
 0: FOR N=0 TO 7: BORDER N: PRINT AT 0,1
4; PAPER N; INK 8;"HIT!": BEEP .1,N: NEX
T N
2045 PRINT AT 0,14; OVER 0;"     "
2050 LET SC=SC+(30 AND C=1)+(50 AND B=1)
+(70 AND D=1)+(100 AND Q=1)
2054 IF SC/1000=INT (SC/1000) THEN  LET
AMMO=AMMO+15
2055 LET HIT=1
2060 BORDER 1: RETURN
2200 LET G$="       GAME OVER": FOR N=5 T
O 5+10: PRINT AT 4,N+5;"e": PRINT AT 4,5
;G$(16-N TO ): PAUSE 5: NEXT N
2205 GO SUB 5020
2210 FOR N=-10 TO 10: BEEP .05,N: BEEP .
05,-N: NEXT N
2220 BORDER 4
2230 CLS : IF SC<=HS THEN  GO TO 2300
2235 LET HS=SC
2240 PRINT "YOU HAVE GOT THE HIGH SCORE!
": INPUT "INPUT YOUR NAME"; LINE N$
2250 IF LEN N$<1 OR LEN N$>15 THEN  PRIN
T "PLEASE RETYPE": GO TO 2240
2300 FOR N=0 TO 7: BORDER N: BRIGHT 1: P
APER N: CLS : NEXT N
2310 PRINT "THE HIGH SCORE IS ";HS;" BY"
'N$
2320 INPUT "ANOTHER GAME ?"; LINE X$
2330 IF X$<>"Y" THEN  STOP
2340 LET INSTR=4030
2345 BORDER 1: PAPER 5: INK 0: CLS
2350 GO TO 20
4000 PRINT AT 10,10;"PLEASE WAIT"
4010 RESTORE 4040: FOR N=USR "A" TO USR
"F"+7: READ A: POKE N,A: NEXT N
```

```
4020 LET SC=0: LET AMMO=20
4022 LET HIT=0
4023 LET C=0: LET B=0: LET D=0: LET Q=0
4025 LET PX=123: LET PY=91
4030 RETURN
4040 DATA 98,55,252,124,56,64,0,0
4050 DATA 2,55,252,120,120,112,32,0
4060 DATA 0,24,36,60,36,36,36,24
4070 DATA 16,16,16,56,56,56,56,56
4080 DATA 0,8,54,65,54,8,0,0
4090 DATA 0,48,124,126,254,252,120,32
5010 OVER 0: PAPER 5: FOR N=1 TO 10: PRI
NT AT N,0;"
   ": NEXT N
5015 OVER 0: PAPER 2: INK 7: PRINT AT 21
,2;"SCORE=0000 AMMO=00 HIGH=0000"
5020 OVER 0: INK 0: PLOT 165,100: DRAW 9
0,75: PLOT 90,100: DRAW -90,75: PLOT 90,
70: DRAW -90,-70: PLOT 165,70: DRAW 90,-
70: PLOT 165,100: DRAW 0,-30: DRAW -75,0
: DRAW 0,30: DRAW 75,0
5100 RETURN
8000 GO SUB 4010: PAPER 5: CLS : LET A$=
"T A R G E T  S H O O T ": FOR N=0 TO 31
8001 PRINT AT 10,(N+1 AND N<31);("a" AND
 N/2=INT (N/2));("b" AND N/2<>INT (N/2))
;AT 10,0; PAPER 2; INK 7; BRIGHT 1;A$(32
-N TO )
8002 PAUSE 5: BEEP .09,N: NEXT N
8003 FOR N=10 TO 2 STEP -1: PRINT AT N,0
;"                                ": PRI
NT AT N-1,0; PAPER 2; INK 7;A$: PAUSE 5:
 NEXT N
8010 PAPER 5: INK 0: CLS
8020 PRINT AT 1,0; PAPER 6;A$
8050 PRINT '" IN THIS GAME YOU ARE LOOKI
NG   DOWN A SHOOTING RANGE WHICH HAS NO
ROOF AND IS THEREFORE OPEN TOTHE SKY."
8060 PRINT '" YOU MUST MANOEUVRE YOUR SI
GHT  (A CROSS) AND TRY TO SHOOT THE  VAR
```

```
IOUS OBJECTS WHICH APPEAR.   SOME WILL M
ATERIALISE AT THE ENDOF THE RANGE, WHILE
 OTHERS WILL FLY OVER THE TOP."
8070 PRINT '" YOU WILL START OFF WITH 20
     ROUNDS OF AMMUNITION, AND EVERY 100
0 POINTS YOU WILL GAIN SOME  MORE."
8075 PRINT '" WHEN YOU HAVE NO AMMUNITIO
N    LEFT, THE GAME ENDS."
8080 INPUT  PAPER 2; INK 9;"PRESS ENTER
TO CONTINUE"; LINE X$
8090 CLS : PRINT ' PAPER 2; INK 7;A$: PR
INT '" USE THE ARROW KEYS TO MOVE YOURSI
GHT, AND THE 'B' KEY TO FIRE."
8100 PRINT ''" A c (CAN) IS WORTH 30 POI
NTS,"'"A d (BOTTLE) IS WORTH 50 POINTS,"
'"A a (DUCK) IS WORTH 70 POINTS"'"AND A
e (CLAY PIGEON) IS WORTH  100 POINTS."
8110 PRINT ''" TO GET YOUR BONUS AMMUNIT
ION   YOU MUST SCORE EXACTLY 1000     PO
INTS, SO THINK CAREFULLY      ABOUT WHAT
 YOU SHOOT AT."
8120 INPUT "PRESS ENTER TO CONTINUE"; LI
NE X$
8130 CLS : RETURN
9990 GO SUB 9900: GO TO 9990
```

Chapter 13

Sliding Puzzle

Description

This is a computerised version of the popular game which involves sliding pieces around a 4×4 grid until the correct picture is obtained. This version provides you with two pictures—a house and a church. Instructions are given below on how to design your own pictures and save them on tape.

Hints on Entry

The program is straightforward to enter and should therefore cause no problems. When the complete program has been

entered, it should be saved on tape or microdrive using the command:

```
SAVE "PUZZLE" LINE 1
```

or

```
SAVE *"M";1;"PUZZLE" LINE 1
```

The original program contains the picture of a house; for variety the data for a church is also included. The data should be entered as lines 9000–9075 in steps of five (it is very important that you do not deviate from this sequence). When entered, this should be saved using the command:

```
SAVE "CHURCH"
```

or

```
SAVE *"M";1;"CHURCH"
```

If you now wish to use the puzzle with the church, simply load the main program, Puzzle and then type:

```
MERGE "CHURCH"
```

or

```
MERGE *"M";1;"CHURCH"
```

and execute the program by typing RUN.

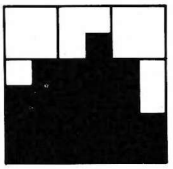
Since you are able to change the picture in such an easy way, there is no reason why you should not create your own pictures for use with the program Puzzle.

The grid is 4 × 4, which means that you need to define sixteen different pieces, including the blank which can be anywhere in the grid and is represented by the data line DATA 0. The data line number for each of the pieces is as follows:

9000	9005	9010	9015
9020	9025	9030	9035
9040	9045	9050	9055
9060	9065	9070	9075

Each piece other than the blank is defined by nine items of data since it is a 3 × 3 square. The numbers in the data statements are the character codes as found in Appendix A on p. 138 of the

Spectrum manual. An example piece would be as shown below with the numerical codes in the grid to the right.



128	132	128
141	143	138
143	143	143

The data line for this piece would be:

DATA 128, 132, 128, 141, 143, 138, 143, 143, 143

The pictures given above use only the sixteen shapes which the Spectrum has available, but there is no reason why you should not use the user-defined graphics to create your own outstanding pictures.

Techniques

This program demonstrates how the MERGE command can be used to exchange data from one program to another.

Many games programs, especially puzzles and word games, require portions of data to store the solution of the problem or the possible combinations which can be used throughout the program. In these cases it would be a waste of time to keep typing in the main program. Also, if the programs are to be stored on a costly microdrive cartridge it would also represent a waste of valuable storage space. A method around this problem is to use the fact that the body of the program would be the same in all cases. So the various sets of data lines can be written as small, separate programs and stored on tape or cartridge, being merged into the main program as and when required.

By using a routine such as that shown below it would be possible to have the merge facility under software control from within the program:

```
8000 REM MERGE CONTROL
8010 INPUT "NAME OF DATA PROGRAM=";A$
8020 INPUT "MICRODRIVE OR TAPE(M/T)";B$
8030 IF B$="M" THEN  MERGE *"M";1;A$
8040 IF B$="T" THEN  MERGE A$
8050 RETURN
```

Instructions

When the program is executed, some instructions are displayed on the screen followed by the grid containing the correct solution. When you have seen this, the solution is replaced by the statement 'creating the problem' while the computer jumbles the pieces. The time taken for this varies, but is usually about one minute.

When the problem is created the grid is displayed with the juggled pieces, and it is your task to solve it in the least number of moves. To move a piece into the empty square you type the letter corresponding to the square you wish to move from. The squares on the grid are represented by the letters A to P as shown on the bottom left of the screen and below.

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P

The problem is solved when the completed picture is shown in the grid.

Listing

IMPORTANT

1) THIS PROGRAM SHOULD BE ENTERED USING CAPS LOCK.

2) SPACES WITHIN TEXT SHOULD BE ENTERED AS LISTED.

House

```
  1 POKE 23658,8
  5 DIM P(16,9): DIM B(16)
 10 INK 0: BORDER 1: LET T=0
 15 PRINT AT 1,4;" ";: FLASH 1: PRINT "
SLIDING PIECES PUZZLE";: FLASH 0
 20 PRINT : PRINT : PRINT "THE IDEA OF
THIS PUZZLE IS TO  RECREATE A PICTURE I
```

```
N A 4 X 4   GRID BY SLIDING PIECES INTO
THE ONE EMPTY SQUARE."
  25 PRINT : PRINT "THIS IS ACHIEVED BY
TYPING THE  LETTER OF THE SQUARE FROM WH
ICH YOU WISH TO MOVE."
  30  PRINT : PRINT "THE GRID IS REPRESE
NTED BY THE  FOLLOWING LETTERS:"
  35 PRINT
  40 FOR I=1 TO 16
  45 IF I<5 THEN  PRINT AT 15,12+I;CHR$
(64+I): GO TO 65
  50 IF I<9 THEN  PRINT AT 16,8+I;CHR$ (
64+I): GO TO 65
  55 IF I<13 THEN  PRINT AT 17,4+I;CHR$
(64+I): GO TO 65
  60 PRINT AT 18,I;CHR$ (64+I): GO TO 65
  65 NEXT I
  70 PRINT AT 20,8;" ";: FLASH 1: PRINT
"PRESS ANY KEY";: FLASH 0
  75 IF INKEY$="" THEN  GO TO 75
  80 CLS
  85 PAPER 5: GO SUB 1000: GO SUB 1300:
GO SUB 1100
  90 PRINT AT 4,1;"THIS IS"
  95 PRINT AT 5,1;"THE"
 100 PRINT AT 6,1;"SOLUTION"
 110 FOR P=1 TO 16
 112 LET B(P)=P
 115 LET R=P
 116 IF R=BLANK THEN  LET SPACE=P: GO TO
 125
 120 GO SUB 1200
 125 NEXT P
 130 FOR I=1 TO 500: NEXT I
 135 PAPER 5: CLS
 140 GO SUB 1400
 145 GO SUB 1300: GO SUB 1100
 150 FOR P=1 TO 16
 155 LET R=B(P)
```

```
 160 IF R=BLANK THEN  LET SPACE=P: GO TO
 170
 165 GO SUB 1200
 170 NEXT P
 175 INK 0: PAPER 5: PRINT AT 5,1;" NUMB
ER": PRINT AT 6,1;"OF MOVES": PRINT AT 8
,6;"0"
 180 LET B$=INKEY$: IF B$="" THEN  GO TO
 180
 185 IF CODE B$<65 OR CODE B$>80 THEN  G
O TO 180
 190 GO SUB 1500: IF L=0 THEN  GO TO 180
 195 LET P=CODE (B$)-64: GO SUB 1600
 200 LET P=SPACE: LET R=B(CODE (B$)-64):
 GO SUB 1200
 205 LET S=SPACE: LET SPACE=CODE (B$)-64
: LET Z=B(SPACE): LET B(SPACE)=B(S): LET
 B(S)=Z
 210
 215 LET T=T+1
 216 INK 0: PAPER 5
 220 IF T<10 THEN  PRINT AT 8,6;T: GO TO
 250
 225 IF T<100 THEN  PRINT AT 8,5;T: GO T
O 250
 230 IF T<1000 THEN  PRINT AT 8,4;T: GO
TO 250
 235 STOP
 250 GO SUB 1700
 260 GO TO 180
 450 RETURN
 998 PAPER 7: INK 0
 999 STOP
1000 FOR J=1 TO 16
1005 PRINT "
     "
1010 FOR I=1 TO 9
1020 READ P(J,I)
1025 IF P(J,1)=0 THEN  LET BLANK=J: GO T
```

```
0 1040
1030 NEXT I
1040 NEXT J
1050 FOR I=1 TO 6: PRINT "
                    ": PAUSE 6: NEXT I
1100 INK 0
1105 FOR I=0 TO 16
1110 PRINT AT 4,10+I;CHR$ 143
1115 PRINT AT 8,10+I;CHR$ 143
1120 PRINT AT 12,10+I;CHR$ 143
1125 PRINT AT 16,10+I;CHR$ 143
1130 PRINT AT 20,10+I;CHR$ 143
1135 PRINT AT 4+I,10;CHR$ 143
1140 PRINT AT 4+I,14;CHR$ 143
1145 PRINT AT 4+I,18;CHR$ 143
1150 PRINT AT 4+I,22;CHR$ 143
1155 PRINT AT 4+I,26;CHR$ 143
1160 NEXT I
1165 RETURN
1200 LET X=5+4*INT ((P-1)/4)
1205 LET Y=7+4*(P-4*INT ((P-1)/4))
1210 INK 4: PAPER 6
1215 PRINT AT X,Y;CHR$ P(R,1)
1220 PRINT AT X,Y+1;CHR$ P(R,2)
1225 PRINT AT X,Y+2;CHR$ P(R,3)
1230 PRINT AT X+1,Y;CHR$ P(R,4)
1235 PRINT AT X+1,Y+1;CHR$ P(R,5)
1240 PRINT AT X+1,Y+2;CHR$ P(R,6)
1245 PRINT AT X+2,Y;CHR$ P(R,7)
1250 PRINT AT X+2,Y+1;CHR$ P(R,8)
1255 PRINT AT X+2,Y+2;CHR$ P(R,9)
1260 RETURN
1300 INK 0: PRINT AT 1,4;" ";: FLASH 1:
PRINT "SLIDING PIECES PUZZLE";: FLASH 0
1310 PRINT AT 15,1;"THE GRID"
1320 FOR I=1 TO 16
1325 IF I<5 THEN  PRINT AT 17,2+I;CHR$ (
64+I): GO TO 1340
1330 IF I<9 THEN  PRINT AT 18,I-2;CHR$ (
```

```
64+I): GO TO 1340
1334 IF I<13 THEN  PRINT AT 19,I-6;CHR$
(64+I): GO TO 1340
1337 PRINT AT 20,I-10;CHR$ (64+I): GO TO
 1340
1340 NEXT I
1350 RETURN
1400 INK 0: PRINT AT 10,6;"CREATING THE
PROBLEM"
1405 FOR J=1 TO 56
1410 LET B=1+INT (16*RND)
1420 LET B$=CHR$ (B+64): GO SUB 1500
1425 IF L=0 THEN  GO TO 1410
1426 IF J<25 THEN  PRINT AT 8,3+J;"*": G
O TO 1430
1427 IF J<29 THEN  PRINT AT J-16,27;"*":
 GO TO 1430
1428 IF J<52 THEN  PRINT AT 12,56-J;"*":
 GO TO 1430
1429 PRINT AT 64-J,4;"*"
1430 LET S=SPACE: LET SPACE=B: LET Z=B(S
PACE): LET B(SPACE)=B(S): LET B(S)=Z
1435 NEXT J
1440 CLS
1500 LET L=0
1501 IF CODE (B$)-64=SPACE THEN  RETURN
1505 IF SPACE=1 AND (B$="B" OR B$="E") T
HEN  LET L=1: RETURN
1510 IF SPACE=2 AND (B$="A" OR B$="C" OR
 B$="F") THEN  LET L=1: RETURN
1515 IF SPACE=3 AND (B$="B" OR B$="D" OR
 B$="G") THEN  LET L=1: RETURN
1520 IF SPACE=4 AND (B$="C" OR B$="H") T
HEN  LET L=1: RETURN
1525 IF SPACE=5 AND (B$="A" OR B$="F" OR
 B$="I") THEN  LET L=1: RETURN
1530 IF SPACE=6 AND (B$="B" OR B$="E" OR
 B$="G" OR B$="J") THEN  LET L=1: RETURN
1535 IF SPACE=7 AND (B$="C" OR B$="F" OR
```

```
B$="H" OR B$="K") THEN  LET L=1: RETURN
1540 IF SPACE=8 AND (B$="D" OR B$="G" OR
 B$="L") THEN  LET L=1: RETURN
1545 IF SPACE=9 AND (B$="E" OR B$="J" OR
 B$="M") THEN  LET L=1: RETURN
1550 IF SPACE=10 AND (B$="F" OR B$="I" O
R B$="K" OR B$="N") THEN  LET L=1: RETUR
N
1555 IF SPACE=11 AND (B$="G" OR B$="J" O
R B$="L" OR B$="O") THEN  LET L=1: RETUR
N
1560 IF SPACE=12 AND (B$="H" OR B$="K" O
R B$="P") THEN  LET L=1: RETURN
1565 IF SPACE=13 AND (B$="I" OR B$="N")
THEN  LET L=1: RETURN
1570 IF SPACE=14 AND (B$="J" OR B$="M" O
R B$="O") THEN  LET L=1: RETURN
1575 IF SPACE=15 AND (B$="K" OR B$="N" O
R B$="P") THEN  LET L=1: RETURN
1580 IF SPACE=16 AND (B$="L" OR B$="O")
THEN  LET L=1: RETURN
1590 RETURN
1600 LET X=5+4*INT ((P-1)/4)
1610 LET Y=7+4*(P-4*INT ((P-1)/4))
1620 PAPER 5
1625 PRINT AT X,Y;"    "
1630 PRINT AT X+1,Y;"    "
1635 PRINT AT X+2,Y;"    "
1640 PAPER 5
1650 RETURN
1700 FOR I=1 TO 16
1710 IF B(I)<>I THEN  RETURN
1720 NEXT I
1730 PAPER 6: INK 2: PRINT AT 12,2;"";
FLASH 1: PRINT "SOLVED";: FLASH 0
1740 STOP
9000 DATA 0
9005 DATA 128,128,128,128,135,128,128,13
3,128
```

```
9010 DATA 128,128,128,128,128,137,128,12
8,143
9015 DATA 137,128,128,128,128,128,128,12
8,128
9020 DATA 128,128,141,128,141,143,128,14
3,143
9025 DATA 143,143,143,143,143,143,128,12
8,143
9030 DATA 143,143,143,143,143,143,143,12
8,128
9035 DATA 142,128,128,143,142,128,143,14
3,128
9040 DATA 128,143,143,128,143,143,128,14
3,128
9045 DATA 128,128,143,143,143,143,128,14
3,128
9050 DATA 143,128,128,143,143,143,128,14
3,128
9055 DATA 143,143,128,143,143,128,128,14
3,128
9060 DATA 128,143,128,128,143,143,128,12
8,128
9065 DATA 128,143,128,143,143,140,128,12
8,128
9070 DATA 128,143,128,140,143,143,128,12
8,128
9075 DATA 128,143,128,143,143,128,128,12
8,128
```

Church

```
9000>DATA 128,128,128,128,128,141,128,14
1,143
9005 DATA 128,128,128,142,128,128,143,14
2,128
9010 DATA 128,128,128,128,128,128,128,14
1,142
9015 DATA 0
```

```
9020 DATA 128,143,139,128,139,133,128,14
2,129
9025 DATA 135,143,128,128,135,128,130,14
1,128
9030 DATA 128,135,139,128,128,128,128,12
8,128
9035 DATA 128,128,128,128,136,128,129,13
9,128
9045 DATA 128,143,143,128,143,143,128,14
3,130
9050 DATA 143,143,141,143,143,143,129,14
3,130
9055 DATA 143,143,143,143,143,143,129,14
3,130
9060 DATA 143,142,128,143,143,128,129,14
3,128
9065 DATA 128,143,128,128,143,140,128,12
8,128
9070 DATA 128,143,128,140,143,143,128,12
8,128
9075 DATA 128,143,128,143,143,143,128,12
8,128
9080 DATA 128,143,128,143,143,128,128,12
8,128
```

GREENWICH LIBRARIES

Other titles available from Micro Press:

THE MICRO MAZE: A GUIDE TO PERSONAL COMPUTING
Wynford James
0 7447 0000 0

20 GAMES FOR THE ORIC-1
Wynford James
0 7447 0003 5

QUALITY PROGRAMS FOR THE BBC MICRO
Simon
0 7447 0001 9

QUALITY PROGRAMS FOR THE BBC MICRO (Cassette)
Simon
0 7447 0011 6

15 GRAPHIC GAMES FOR THE SPECTRUM
Richard G. Hurley
0 7447 0002 7

MASTERING THE TI-99
Peter Brooks
0 7447 0008 6

ADVANCING WITH THE ELECTRON
Peter Seal
0 7447 0012 4

QUALITY PROGRAMS FOR THE ELECTRON
Simon
0 7447 0004 3

MAKING THE MOST OF YOUR SPECTRUM MICRO DRIVES
Richard G. Hurley
0 7447 0005 1

GRAPHIC ADVENTURES FOR THE SPECTRUM 48K
Richard G. Hurley
0 7447 0013 2

THE SPECTRUM OPERATING SYSTEM
Steve Kramer
0 7447 0019 1

EDUCATIONAL GAMES FOR THE BBC MICRO
Ian Soutar
0 7447 0016 7

THE ATMOS BOOK OF GAMES
Wynford James
0 7447 0018 3

THE COMMODORE 64 BOOK OF SOUND AND GRAPHICS
Simon
0 7447 0015 9

QL SUPERBASIC: A PROGRAMMER'S GUIDE
John Wilson
0 7447 0020 5

THE QL HANDBOOK
P.C. Sturgess and
M.J. Laverick
0 7447 0021 3

THE QL BOOK OF GAMES
Richard G. Hurley
0 7447 0022 1

SPECTRUM SUPERGAMES

Here are thirteen SUPERGAMES designed specifically for the Spectrum 16K and 48K to test your skill and ingenuity!

FANTASTIC JOURNEY takes you on a superb adventure, and MISSILE ATTACK tests your ability to defend your city from threatened annihilation. Keep a straight face while playing CRIBBAGE against the computer and improve your hand-to-eye co-ordination with CANYON BOMBER, MADCAP RALLY and ROAD RACER. Have you got a steady hand? Then test your skill against your friends with SHOOTOUT.

Each program is accompanied by a description of the techniques used in games programs. High score routines, joystick control, memory saving methods and modular programming are described using the listings as examples to help you improve your own programming.

The Author
Richard Hurley read Astrophysics at the University of London, UK, and is now Head of the Computer Studies Department at Hurstpierpoint College, UK. He is an experienced programmer and is the author of several books on the Spectrum.

GB £ NET +005.95
ISBN 0-7447-0017-5
00595
9 780744 700176